The self-excited vocoder for mobile telephony

Hudson, Nicholaus D. W.

Award date:
1992

Awarding institution:
University of Bath

THE SELF-EXCITED VOCODER FOR MOBILE TELEPHONY

submitted by
Nicholaus D.W. Hudson
for the degree of PhD.
of the University of Bath
1992

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Summary

The considerable growth in the use of cellular telephones over the past decade, coupled with future anticipated demands and the move to digital speech coding, has prompted much research into speech coding at rates below 8kbit/s. This thesis reports a three year study into the use of the Self-Excited Vocoder (SEV) for this application.

In the SEV, theoretically, the synthesis filter excitation is derived entirely from the previous history of the excitation. In practice, this is achieved using a conventional Linear Prediction Long Term Predictor (LTP), with no input. This is termed the Self-Exciting LTP (SE-LTP). It is demonstrated, in this thesis as by other researchers that, after initialisation, excellent speech quality can be achieved. However, such a simplistic scheme has very poor inherent error robustness and in a practical mobile telephony scheme, channel errors are generally unavoidable.

The perceptually weighted, analysis-by-synthesis SE-LTP is studied in detail. Initialisation techniques which combine fixed and adaptive portions of the adaptive codebook are used to enable error robust performance. The adaption mechanism of the codebook is modified which further improves the error robustness, as well as, producing a more perceptually pleasing characteristic distortion. The investigation makes regular use of both objective and subjective testing and comparison of self-excited with CELP techniques.

The thesis also incorporated a study of multi-predictor SEVs and investigates the performance of established Series and Parallel SEVs. The outcome of the research is the introduction of a hybrid combination, termed the Series/Parallel SEV, which possesses two beneficial features namely the excellent clear channel performance of the Series SEV and the excellent error robustness of the Parallel SEV.

Acknowledgements

The author wishes to express his thanks to his supervisor Dr. R.J. Holbeche and to the Science and Engineering Research Council and Vodafone Ltd for their financial support throughout this work. Also special thanks are expressed to the many people who assisted with the listening tests.

Contents

Glossary

BER	Bit Error Rate
ACELP	Adaptive Code Excited Linear Predictive (Vocoder)
CELP	Code Excited Linear Predictive (Vocoder)
GMSK	Gaussian Minimum Shift Keying
GSM	Group Spécial Mobile
LAR	Log-Area Ratio
LPC	Linear Predictive Coding
LTP	Long-Term Predictor
RELP	Residually Excited Linear Predictive (Vocoder)
RF	Radio Frequency
RPE	Residual Pulse Excited (Vocoder)
MPE	Multi-Pulse Excited (Vocoder)
PCM	Pulse Code Modulation
PFC-SEV	Partial Fixed Codebook Self-Excited Vocoder
PSTN	Public Switched Telephone Network
SE-LTP	Self-Exciting Long-Term Predictor
SFC-SEV	Separate Fixed Codebook Self-Excited Vocoder
SNRSEG	Segmental Signal to Noise Ratio [37]

Chapter 1

Introduction

Chapter 1

Introduction

The increased mobility of life over the last few decades has inevitably brought with it the requirement for improved communications. The need to communicate "on the move" has inspired a rapidly increasing world-wide radio communications industry. This initially centred around the police and fire services, although post world war II saw the introduction of simple private and business systems. Since then, there has been a never ending demand for more channels, more facilities and improved performance in the U.K.'s land mobile radio services [19].

More recently, mobile communications systems have become even more widespread, including pagers, cordless telephones, analogue cellular telephones and Telepoint, the first digital public cordless telephone system. 1992 has seen the introduction of GSM, in Great Britain, a Pan European digital mobile telephone system. 1993 will see the introduction of PCN, hoping to bring mobile telephony to the mass market. These new systems see the introduction of digital speech transmission and have prompted much research into coding techniques.

1.1 Cellular Radio

The cellular concept is reported to have been discussed at the Bell telephone Laboratories as long ago as 1947 [55]. It was always clear that, while the system concept was simple, the implementation would be difficult. There were many reasons for this: The required technology, particularly the computing elements, was not available until recently. The necessary chunks of spectrum suitably spaced for duplex operation could only be found in a high part of the spectrum not then used for mobile communications. Another major reason was the high investment needed, this factor has decreased in significance with the world wide increase in fuel costs and the realisation of the possible financial rewards from the operation of such a system.

The first cellular systems were in the United States, with their Advanced Mobile Phone System (AMPS) [55], in Japan with their High Capacity Automobile Telephone System [26] and the Nordic countries with their Project Nordic Mobile Telephone [36]. The British Total Access Communication System (TACS) came into service in January 1985 and was based on the American AMPS system.

1.1.1 TACS

To date, the U.K. has two separate cellular radio networks, operated by Cellnet and Vodafone (formerly Racal Vodafone). From their launch in 1985 both have expanded rapidly and offer communication from over 97% of the country to anywhere in the world, its advantages have been quickly taken up by people from senior company executives to delivery drivers. This quality of service is not inexpensive, of course, but many companies find the improved employee efficiency well worth the investment.

The demand for service, particularly in central London, together with the limited frequency spectrum available has necessitated techniques for significantly increasing cell numbers. These include 120° antenna sectoring, where one radio site can serve 3 cells, and additional small coverage low power cells. The frequency band has also been extended, with the ETACS standard, to help cope with the increased number of users. However the likely demand into the next century cannot be met by the spectrum capacity of both U.K. TACS. cellular telephone systems. New mobile telephone systems are required in addition to these two existing systems to cope with the anticipated demand. The two TACS systems will remain operational well into the 21st century and will work alongside future digital systems.

1.1.2 GSM

At present there are six different analogue cellular standards in use in Europe. Each is likely to reach saturation within the next few years. A cellular user is unable to roam

across national boundaries, and production volumes for cellular phones and infrastructure equipment are restricted, leading to higher prices for users and limiting the export potential of European cellular technology.

In 1982, under control of the Conférence Européene des Administrations des Postes et des Télécommunications (CEPT) the Groupe Spécial Mobile (GSM) was tasked with the drawing up of a common European specification for the next generation of European cellular telephone equipment. To date, the most complex radio system ever has been devised. 1990 saw agreement over a basic set of 121 recommendations covering all aspects from terminal to infrastructure design. Including supporting documentation, the specification consists of over 6000 pages. As the full introduction draws nearer, the abbreviation GSM has changed from that of the spearheading committee, to the "Global System for Mobile Communications."

GSM will use five different classes of mobile ranging from the 20W class 1 vehicle model down to the 0.8W class 5 hand-held terminal. Greater emphasis will be placed on coverage for the hand-held, this being a lesson learned from previous generation systems, which were optimised for use with high powered vehicle mounted terminals.

The most notable decision by the GSM committee was to opt for digital speech transmission. System performance is higher at the price of greater system complexity. In cellular radio, most important is spectral efficiency, ie the number of users that can be supported, for a given bandwidth, within a given geographical area. Both GSM and TACS support one communication channel in 25kHz of bandwidth. The digital signal processing technology within GSM is designed to operate at carrier to interference (C/I) ratio of 10-12dB, compared to 17-18dB for TACS. This factor will allow smaller reuse distances for channels which is expected to give GSM up to a three-fold improvement in spectral efficiency [5]. Looking further into the future, facility exists for incorporation of a half-rate speech codec without modification to the radio subsystem, doubling the

number of voice channels per carrier, giving GSM something like a five-fold improvement in spectral efficiency. The planned date for the introduction of this half rate speech coder is 1995.

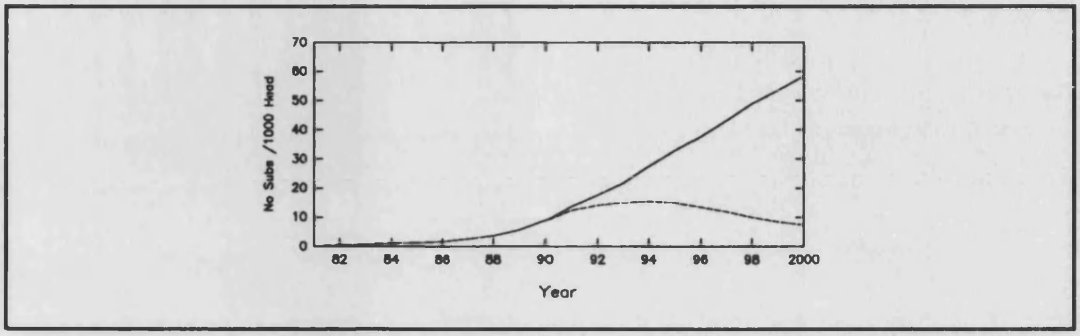

Figure 1.1 *Forecast for European GSM Market Growth, Dotted Line Analogue, Solid Line GSM + Analogue. (After [11]).*

Figure 1.1 shows a recent forecast of cellular subscribers in western Europe [11] per 1000 head of population. This translates to a European market of 18.4 million subscribers by the year 2000. British operators Cellnet and Vodafone aim to provide GSM networks that match the regional coverage of their analogue networks by 1994, however along with many of their European counterparts their marketing policy will have to satisfy two goals, the desire to use the new GSM capacity and preservation of the investment value of their analogue networks.

1.1.3 PCN

1992 sees the scheduled launch of the "Personal Communications Networks" (PCN), intending to bring mobile communications to the mass market. Licences were awarded, in December 1989, to three consortia: Mercury, Unitel and Microtel. The PCN operators have adopted the slogan "Phones for people not places!" and intend to compete with both existing cellular and with public switched fixed network services. The intention being that a user will have just one handset, not a separate mobile and fixed phone.

150MHz of spectrum have been allocated to PCN compared to 50MHz for GSM. The PCN system will operate at twice GSM frequencies (around 1800MHz) and will operate

with low power mobiles of two power classes: 1W and 250mW. At this higher frequency, propagation losses increase by 6-8dB and coupled with low mobile output powers leads to small cell sizes, consistent with supporting a mass market. Estimates give GSM national coverage with about 700 base stations whereas PCN will require between 4000 and 10000 for near equivalent coverage [11].

The European Telecoms Standards Institute (ETSI) has developed the PCN standard known as DCS1800. This is almost entirely based upon the GSM standard, the only adaptations arise from operation in the 1800MHz region and across a much wider frequency band.

1.2 Speech Coding For Digital Cellular Telephony

The underlying goal of digital speech encoding, is to transmit speech, with the highest possible quality, over the least possible channel capacity, and with the least cost. As can be expected, the cost of speech encoding increases with coder complexity, which in turn increases with code efficiency and channel utilisation. A further problem is designing a speech and associated channel coder to survive bursty, impulsive interference typical of the digital cellular telephone domain.

Speech coders can be divided into two classes, Waveform coders and Source coders (Vocoders).

A waveform coder attempts to reproduce the signal waveform. This principle makes operation signal independent, hence they can encode equally well a variety of signals, speech, music, tones or voiceband data. They also tend to be robust for a wide range of talker characteristics and for noisy environments. To obtain these advantages with minimum complexities, waveform coders typically aim for moderate economies in transmitted bit rate. Waveform coders are typically optimised for greater coding efficiency by observing statistics of a certain signal class (ie speech) so performance is optimised for this type of signal.

A source coder depends for operation on a knowledge of how the signal was generated at source. This requires that the signal must be fitted into a specific mould (in this case speech) and parameterised accordingly. Source coders for speech are generally referred to as Vocoders (*voice coders*), and give a synthetic speech quality from a very low data rate.

A hybrid coder is a combination of source and waveform coders. These are the main area of interest within speech coding research. Not surprisingly these have data rates between those of the source coders and the waveform coders with the prospect of very good perceptual speech quality.

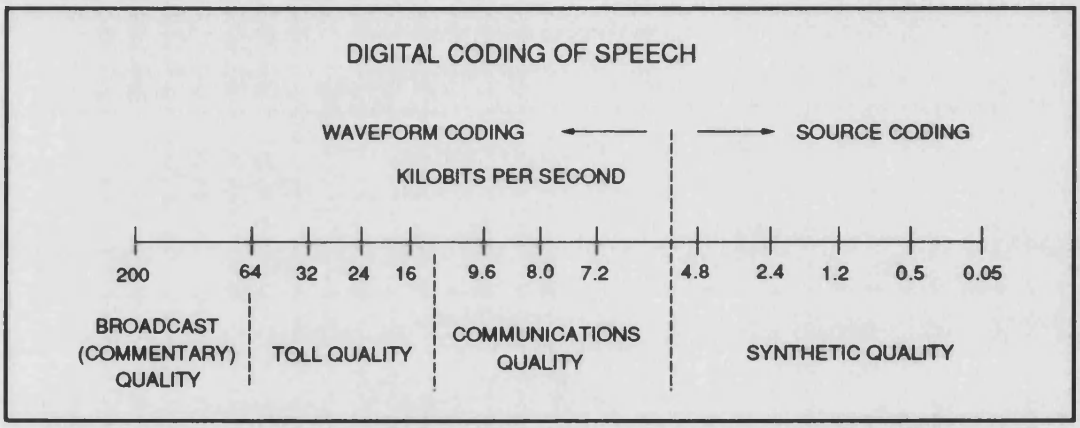

Figure 1.2 *The Range of Speech Coding Transmission Rates (Nonlinear Scale) and Associated Quality, (After [14]).*

Figure 1.2 shows the cross-section of bit-rates of interest in speech encoding. The figure highlights the higher bit-rate requirements for non speech-specific waveform coders as opposed to speech specific vocoders. The figure also indicates the speech quality capability at a given bit-rate, the characterisations are referred to as commentary, toll, communications and synthetic. For telecommunications, toll quality is required. This implies quality comparable to an analogue speech signal having approximate properties: Frequency range 300-3400Hz; Signal-to-noise ratio ≥30dB; Total harmonic distortion

≤2-3%. Toll quality is already achievable with low complexity coders operating at bit rates of around 13kbit/s. Present research is aimed at producing toll quality speech transmission at around half this figure. This has far greater computational demands.

Current toll quality speech coders operating at around 16kbit/s have relatively low computational complexity. Future coders operating at 8kbit/s or less will require significantly more processing power and significantly more complex algorithms. Development of a speech coder is then split into two stages: Firstly, development of the speech coding algorithm, by simulation in non real time. Secondly, real time implementation using custom integrated circuits or DSP devices with highly optimised software. The Self-Excited Vocoder research described within this thesis develops the speech coder algorithm.

1.3 The Self-Excited Vocoder

The Self-Excited Vocoder was introduced in 1986 by Rose and Barnwell [43]. It is another member of the family of linear predictive speech coders. This thesis describes a three year study into the use of this vocoding technique as a codec for mobile telephony use at bit rates below 8kbit/s

This thesis demonstrates, both in low complexity versions and high complexity versions that, after initialisation, the SEV is capable of excellent speech quality. However, this simplistic scheme has appalling inherent error robustness and in a practical mobile telephony scheme, channel errors are generally unavoidable.

The perceptually weighted, analysis-by-synthesis SE-LTP is studied in detail in chapter 6. Initialisation techniques which combine fixed and adaptive portions of the adaptive codebook are used to enable error robust performance. The adaption mechanism of the codebook is modified which further improves the error robustness, as well as, producing

a more perceptually pleasing characteristic distortion. The investigation makes regular use of both objective and subjective testing and comparison of self-excited with CELP techniques.

Chapter 7 moves on to study multi-predictor SEVs and investigates the performance of established Series and Parallel SEVs. The outcome of the research is the introduction of a hybrid combination, termed the Series/Parallel SEV, which possesses two beneficial features namely the excellent clear channel performance of the Series SEV and the excellent error robustness of the Parallel SEV. The development of speech coders within this thesis is shown in figure 1.3.

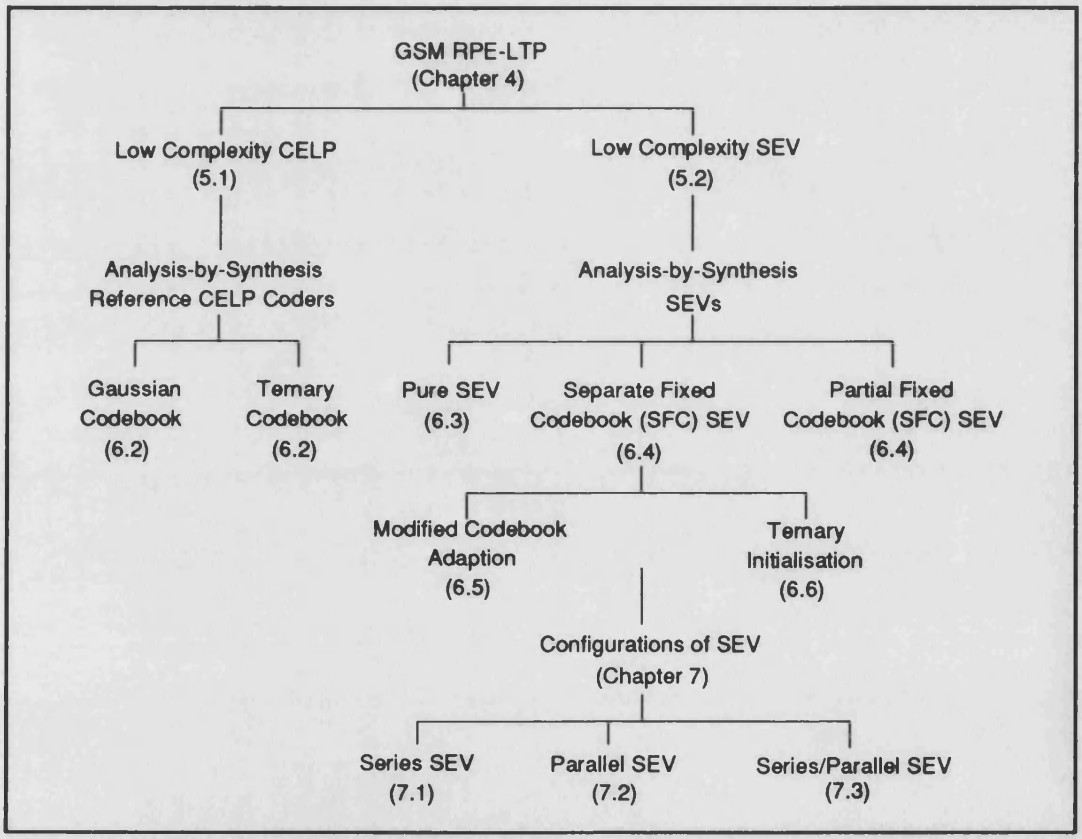

Figure 1.3 *The Development of Speech Coders within this Thesis. Figures in Parenthesis Correspond to Chapter and Section Number.*

Chapter 2

Human Speech and Hearing

Chapter 2

Human Speech and Hearing

The earth is abuzz with numerous languages and dialects, each and every one being produced by the same vocal apparatus. This speech is an analogue signal representing linguistic information described by discrete phonetic symbols. In addition, both meaning and mood can be added to an utterance. By altering voice pitch, a phrase can be emphasised or a statement changed into a question.

This chapter is intended to outline the important features of human speech that should be appreciated by the speech-coding researcher. It must be stressed that it is difficult make concise, accurate statements about human speech and its perception. However, some generalised ideas are useful in the study of speech coding.

Section 2.1 describes the human speech apparatus. Section 2.2 describes the physical features of the speech waveform, differentiating between voiced and unvoiced speech, discussing frequency spectrum and fundamental frequency. Section 2.3 discusses the acoustic features of the speech waveform, distinguishing between vowels and consonants, and also introducing the idea of "speaking speed". Section 2.4 moves on to discuss the hearing aspects of speech communication and covers the hearing apparatus. The final section, 2.5, discusses speech perception and the current limitations in its understanding.

2.1 Basic Physical Principles

Human speech is an acoustic pressure wave resulting from complex voluntary movements of the human speech apparatus, depicted in figure 2.1. Air is expelled from the lungs into the trachea and then forced between the vocal folds into the vocal tract. Periodic opening and closing of the vocal folds can pulsate the airflow at a rate dependant upon the air pressure in the trachea and the instantaneous adjustment of the vocal folds.

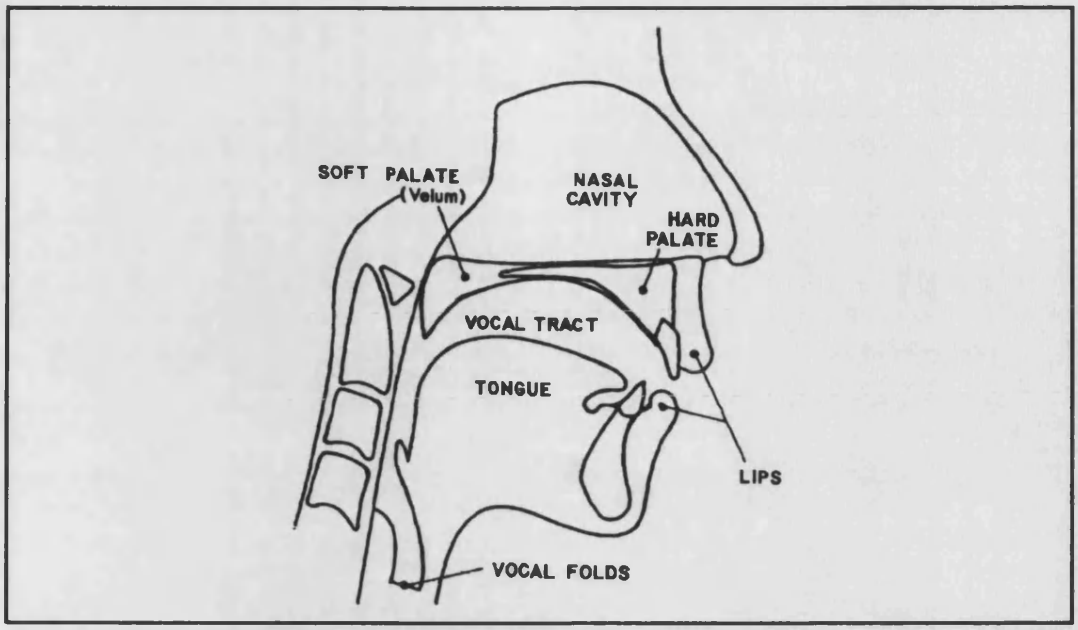

Figure 2.1 *Human Speech Production Apparatus.*

The higher their tension, the higher the perceived pitch or fundamental frequency of the sound. Sounds produced in this manner are termed voiced, for example the phonetic /i/ in *e*ve.

An unvoiced sound is generated by voluntarily holding the vocal folds open and constricting the airflow in the vocal tract, causing turbulence, generating acoustic noise. An example of this is the /f/ in *f*ish where the constriction results from setting upper teeth on lower lip. A voiced fricative, such as /v/ in *v*an, is generated with both a constriction and a vocal fold vibration. A plosive sound, such as /p/ in *p*op, is generated by building up air pressure in the mouth and then suddenly releasing the air.

Once a source of sound has been established, certain frequency regions are intensified by resonances in the vocal tract. The vocal tract is a non uniform acoustic tube extending from the vocal folds to the lips. The lips, jaw, tongue and velum can vary vocal tract shape as a function of time producing each sound with an individual quality. This process is called articulation.

The nasal cavity is an extra acoustic tube for articulation and sound transmission. It is used in the generation of the nasal sounds such as /n/ and /m/ in ru*n* and ru*m* respectively. Acoustic coupling between the nasal cavity and vocal tract is controlled by the size of the opening at the velum.

2.2 The Speech Waveform

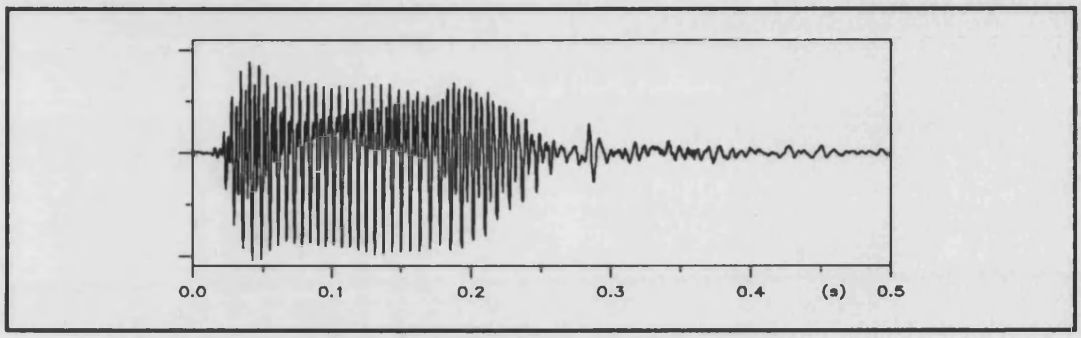

Figure 2.2 *Human Speech Waveform "eve".*

Figure 2.2 shows the speech waveform of the utterance "eve" by a male speaker. The first half of the trace, the /e/, shows considerable voicing, with an average pitch period of 155Hz.

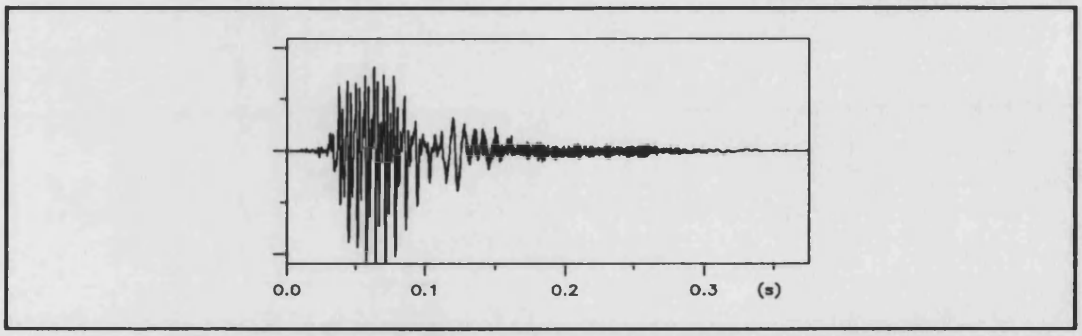

Figure 2.3 *Human Speech Waveform "fish".*

Whereas figure 2.3 shows the waveform of utterance "fish", the first 60ms of the sound, which is the fricative /f/ shows a much more complicated and far less obviously periodic structure. This is a result of combining voiced and unvoiced sound. The last 100ms of this word, the /sh/, is an entirely unvoiced sound and it has a low level noiselike waveform.

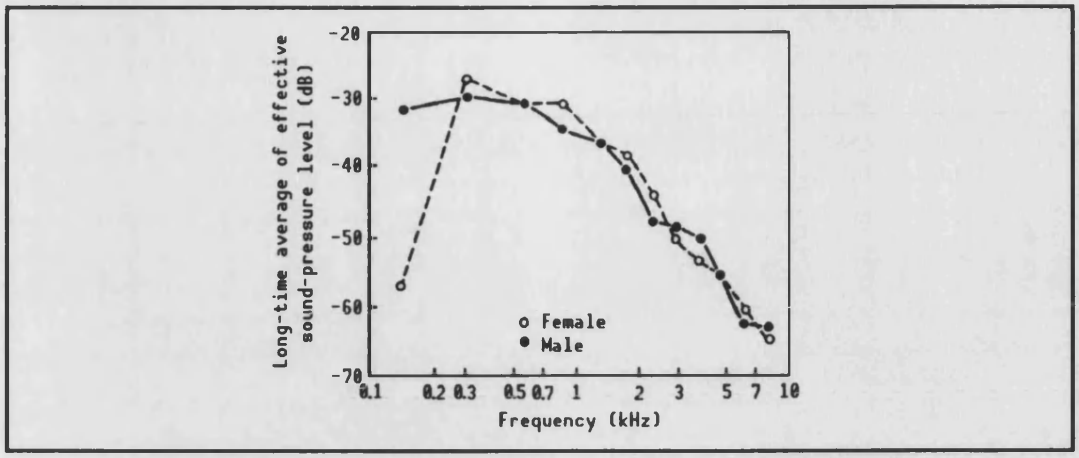

Figure 2.4 *Long-time averaged frequency spectrum of speech: after [44]*

Frequency components and intensity of speech change continuously with time. A long time averaged frequency spectrum is shown in figure 2.4. The only difference between male and female voices appears at the lower end of the frequency scale, where female speech energy is almost null. Differences due to language are hardly noticeable [44]. Almost 80% of the speech energy is contained within frequency components lower than 800Hz. This spectrum can be thought of as being approximately flat up to 800Hz then sloped at -10dB/octave above this.

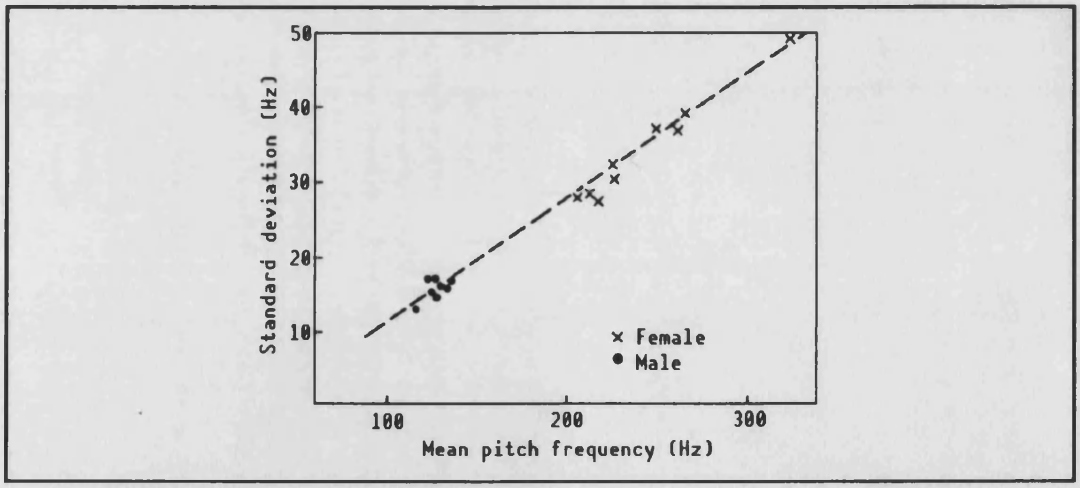

Figure 2.5 *Means and standard deviations of voice pitch frequencies: after [44]*

A speech waveform consists of two parts, the noiselike part in which the amplitude varies randomly and the periodic part which repeats almost the same waveform cyclically. This repeating period is called the fundamental frequency and corresponds to the frequency of vocal cord oscillations.

The fundamental frequency for a conversational speech waveform varies continuously and slowly in time. The average and standard deviations of fundamental frequency for individual speakers are shown in figure 2.5. The deviation for a female voice is twice that of a male voice.

2.3 Acoustic Features of the Speech Waveform

2.3.1 Vowel

A vowel is produced entirely by voiced excitation of the vocal tract. The tract is kept in a relatively stable shape for most of the sound. With a vowel, there is negligible, if any, nasal coupling so sound radiates entirely from the mouth (excepting the small transmission through cavity walls).

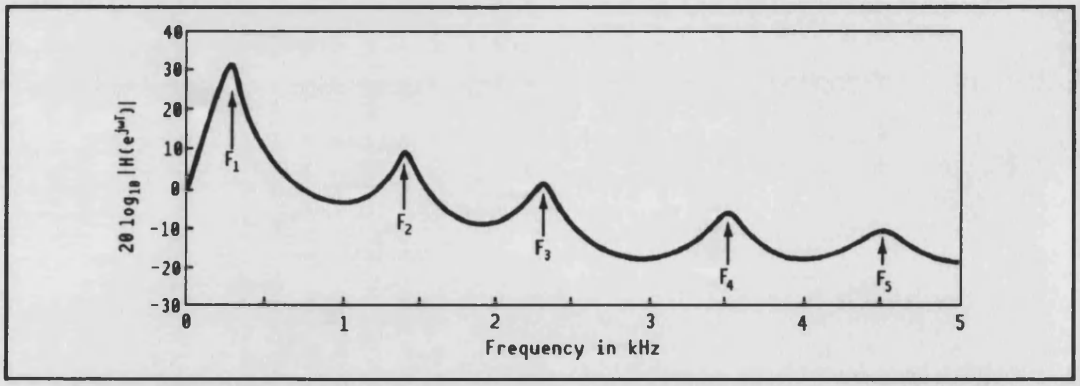

Figure 2.6 *Example of vowel spectral envelope with the formants F_i marked. (After [46])*

An example of the vowel spectral envelope is shown in figure 2.6, it is characterised by a number of formants, or predominant frequency components. They are classified into first, second and third etc, from lowest to highest resonant frequencies. These formants vary depending on the speakers sex and other characteristics as well as being

dependant upon preceding and following phonemes. These variations arise since there is a very close relationship between formant frequency and vocal tract length and shape. The perception of a vowel is most significantly dependant upon the lower formants.

2.3.2 Consonant

Alternatively, the consonants are those sounds which are not entirely voiced, mouth radiated or articulated from a relatively stable vocal tract. They generally have greater vocal tract constrictions than the vowels and can be excited and radiated differently.

A consonant is produced, as a result of articulation, by airflow turbulence generated near the point of constriction of the vocal tract. A stop is produced by the airflow caused by a release of pressure at a closure in the vocal tract. Fricatives are generated at the place of maximum constriction and affricatives are generated in between. A consonant is voiced or voiceless (unvoiced) depending upon whether or not the vocal cords vibrate.

Nasal sounds are radiated from the nostrils through the nasal cavity with the vocal tract closed at the lips or somewhere along its length. In addition, there are sounds called glides and flaps. These sounds are transitional, in contrast to steady vowels. The characteristics of a given phoneme depend upon periodicity of its waveform, frequency spectrum, duration and transition.

Acoustical features vary continuously in between each phoneme during normal conversational speech because of the smooth coupled movement of the vocal organs in the transitional part of each phoneme, This is known as coarticulation. Acoustical features also vary according to the change in adjoining phonemes caused by coarticulation. Thus it is not possible to easily define a phoneme from its acoustic features.

2.3.3 Time Rate

In human sentences, speech and pause intervals alternate. The percentage of the total time occupied by speech is called the time rate. This varies with "speaking speed" and at normal speed it is approximately 68% [44]. The "speaking speed" is adjusted by varying the duration of the pause intervals. This allows some generalisations about phoneme duration to be made. A vowel has a fairly constant duration of about 70*ms*. However the duration of a consonant can vary between 5 and 130*ms* depending upon its type. Thus syllables vary from 75 to 200*ms* giving an average time of 130*ms*.

2.4 Hearing

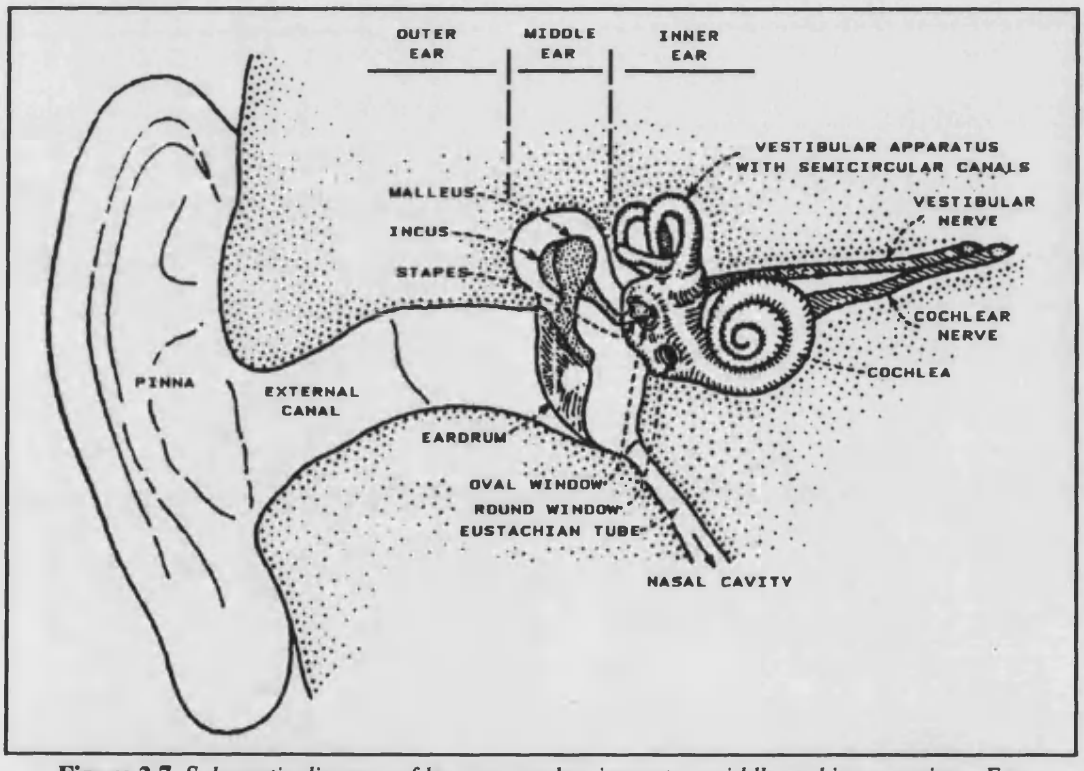

Figure 2.7 *Schematic diagram of human ear showing outer, middle and inner regions. For illustrative purposes the inner and middle ear structures are shown enlarged.*

The final component in any speech communication link is the human ear. For this reason, the speech coding researcher should be familiar with its overall

acoustic-mechanical operation, along with present limits of knowledge of inner ear processes, transmission of neural information to the brain and the mechanism of perception.

The human ear is shown schematically in figure 2.7, it is divided into 3 regions, the outer ear, the middle ear and the inner ear.

2.4.1 Outer Ear

The Pinna, the exterior organ of the human ear has main function of protecting the entrance to the external canal, although it also has directional characteristics at high frequencies enabling localisation of sound sources (this is utilised more fully in some animals). The external canal is about 2.7cm in length, 0.7cm in diameter with a volume of the order of $1cm^3$. At the inner end it is blocked by the eardrum, a thin flexible membrane.

2.4.2 Middle Ear

On the other side of the eardrum lies the middle ear containing the ossicular bones, the malleus, incus and stapes. The malleus is fixed to the eardrum, it makes contact with the incus, which in turn connects via a small joint to stapes. The footplate of the stapes seats in a port, the oval window, and is retained by an annular ligament. This oval window is the entrance to the inner ear. The overall function is impedance matching of an external sound pressure from the air medium of the outer ear to the fluid volume displacement of the liquid medium of the inner ear.

The important feature of the middle ear, is its transmission as a function of frequency. The efforts of previous researchers are discussed by Flanagan [13] concluding that results obtained are very different, suggesting the characteristic is a function of sound intensity and that it varies substantially from individual to individual. The common indication is that the middle ear transmission has a low-pass characteristic.

The other purpose of the middle ear, of little interest to the speech coding researcher, is protection against loud sounds which may damage the more delicate inner ear. Muscles attenuate the vibratory amplitude of the eardrum when subject to loud sounds.

2.4.3 Inner Ear

The inner ear consists of the cochlea (normally coiled like a snail shell in a flat spiral of 2.5 turns) and the vestibular apparatus for head motion sensing. It is in the cochlea that auditory mechanical to neural transductions take place.

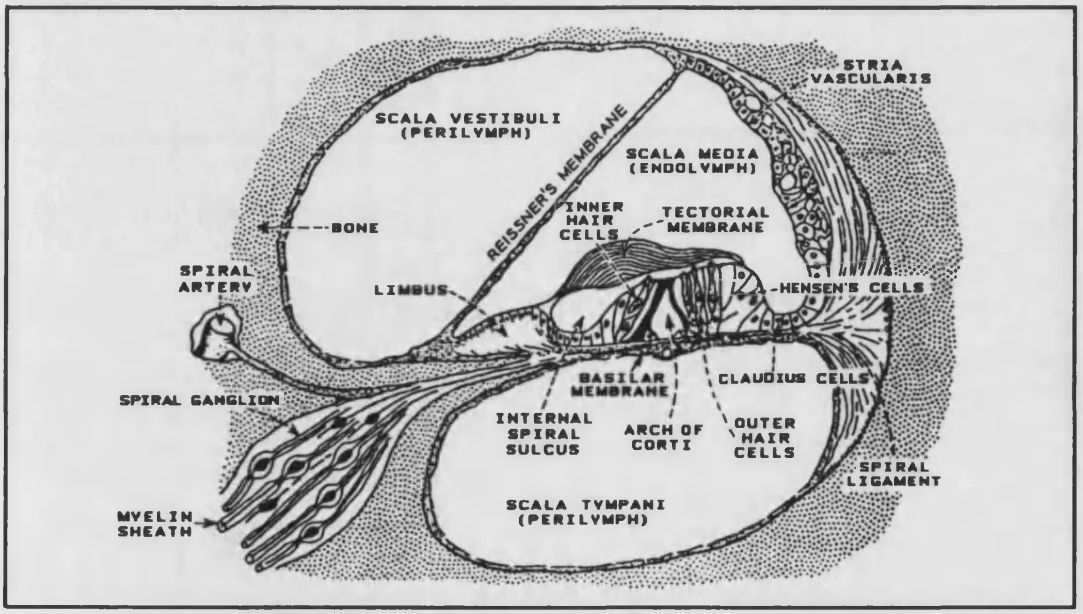

Figure 2.8 *Schematic cross section of the cochlear canal.*

A cross section through the cochlea is shown in figure 2.8 showing the cochlea duct with its three fluid filled channels, the Scala Vestibuli, the Scala Media and the Scala Tympani. The latter two of these channels are separated by the basilar membrane, which supports the organ of Corti where the mechanical to neural transduction is performed by 30000 sensory (or hair) cells.

The cross-sectional area of the cochlea at the stapes end is about $4mm^2$ and this area decreases to about $1mm^2$ at the tip. The basilar membrane is stiffer and less massive at its narrow end, and more compliant and more massive at the broad end. Its resonant

properties therefore vary along its length. Békésy [47] discovered travelling waves on this basilar membrane and realised it could be thought of as a non-linear transmission line. High frequencies travel only a short distance, while low frequencies travel much further, the lower the further, before attenuation.

The amplitude and phase response at a given point on the basilar membrane is similar to that of a broad band-pass filter. The amplitude responses of successive points are roughly constant-Q in nature. Because of this constant percent bandwidth property, the frequency resolution is best at the low-frequency (broad) end and the time resolution is best at the high-frequency (narrow) end. Psychoacoustically, the just noticeable frequency difference between two tones presented in succession to a human listener is less than 3Hz at 1kHz.

Relatively little is known about the mechanism by which the basilar membrane movements are converted into neural activity. Even less is known about how information is coded in nerve pulses and assimilated into auditory percept by the brain. It is known that the hairs of the sensory cells, in experiencing a lateral shear owing to relative motion of the basilar membrane and tectorial membrane, generate local electrical potentials which represent the local basilar membrane displacement.

Many researchers have attempted to mathematically model the operation of the ear. Schroeder describes a number of these models [47], however research in this direction is still in its infancy and no completely satisfactory model exists. Another unanswered question is that of what is known as "periodicity pitch" ie. the sensation of a sound quality without the presence of a physical component in the acoustic stimulus at the perceived frequency. This is often encountered in older telecommunications systems where the lower limit of the frequency range is 300Hz. Fundamental frequency

components are attenuated and only higher harmonics transmitted, requiring human perception to recreate missing fundamental components. This effect continues to challenge psychoacousticains, psysiologists and model builders alike.

2.5 Speech Perception

Human perception is approached from two viewpoints: Firstly the abilities and limitations of the hearing organ as a mechano-neural transducer of all acoustic signals. Secondly the identification and classification of auditory patterns which are significant within the communicative experience of the listener.

Man is highly sensitive to differences in the frequency or intensity of sounds presented for comparison. Under certain conditions, the threshold for detecting a difference in the frequency of two tones may be as small as 1 in 1000 [13]. The threshold for detecting differences in intensity can be less than 1dB. On the basis of comparative judgements, it has been estimated that a typical person can distinguish about 350000 different tones.

Conversely, a listener is unskilled at identifying these attributes in isolation. When equally loud pure tones are listened to individually for absolute judgement of frequency, a typical listener is only able to identify five different tones. The number of correct identifications increases when the sound stimulus is made "multidimensional" ie. by quantising it in frequency, loudness or duration etc, [13].

The personality of a speaker is mainly dependant upon the formant frequencies and bandwidths. It has been shown [27] that personality is strongly sensitive to formant frequency shift, especially to shift of the lower three formants, and it is lost by the uniform shift of 5%. Alternatively the personality is well preserved for a change in formant bandwidth and this must be either widened five times or narrowed to one fifth of its original bandwidth to eliminate the personality.

It should also be noted that personality of a speaker is also included in such linguistic features as manner of speaking and intonation. Consequently longer sentences are not suited to examination of the relationship between vocal tract characteristic and speech personality.

Multiple distortions in the digital encoding of speech have differing effects in the manner in which they influence intelligibility and acceptability. Goodman *et al* [16] conducted experiments on the effects of three distortions: bandwidth reduction, peak clipping and amplitude quantisation. He concluded that the effects of multiple digital impairments on speech quality are not easily predicted from measurements of the effects of single impairments. Also apparent was that intelligibility and acceptability are influenced differently by the multiple distortions.

Another important aspect of hearing is masking, where the perception of one sound is obscured by the presence of another. Simultaneous sounds cause frequency masking, where a lower frequency sound masks a higher frequency one. Sounds delayed with respect to one another can cause temporal masking of one or both sounds. Masking is the major nonlinear effect that prevents considering the perception of speech as the summation of responses of the individual sound components. In speech coding applications, quantisation noise from the coding process can be suitably distributed in frequency to be masked by high speech energy in the formant regions. This is termed "perceptual error weighting".

The ear is relatively insensitive to phase variations, as long as group delay variations are less than a few milliseconds [47]. Randomising the phase angles in the a short-time Fourier transform of speech has less perceptual effect than changing the amplitude spectrum. However, while this time-invariant phase distortion is fairly unimportant perceptually, the naturalness of a speech signal is reduced.

Chapter 3

Linear Prediction of Speech

Chapter 3

Linear Prediction of Speech

It is well known that adjacent speech samples are highly correlated. Firstly this is a result of a resonances in the vocal tract that "ring". Secondly, for voiced speech, each sample is known to be highly correlated with the corresponding sample that occurred one pitch period earlier. For efficient coding of speech, these redundances must be exploited. Redundancy due to the vocal tract resonances is modelled using short-term prediction which is the subject of the first major section of this chapter. Redundancy due to utterance pitch is modelled using long-term prediction which is the subject of the second major section of this chapter. The final major section describes a number of linear predictive speech coders which utilise these predictions.

3.1 Short-Term Prediction

This subsection discusses the theoretical aspects of short-term linear prediction. It begins with the linear all-pole model of human speech production in section 3.1.1. Section 3.1.2 covers the estimation of short-term linear prediction parameters with both the autocorrelation and covariance methods. Section 3.1.3 introduces lattice formulations, which have notable stability advantages. Section 3.1.4 discusses transmission of filter parameters using both log-area ratios and line spectrum pairs, leaving the final section, 3.1.5, to introduce perceptual error weighting for speech coders.

3.1.1 Linear Model of Speech Production

A block diagram of this human speech production model is shown in figure 3.1. The input to this model $e(t)$ is either an impulse train with period p for voiced sounds, or random white noise for unvoiced sounds. At this stage it should be noted that there is no provision for mixing these inputs to simulate voiced frication, or to couple in an extra filter branch to simulate the nasal cavity.

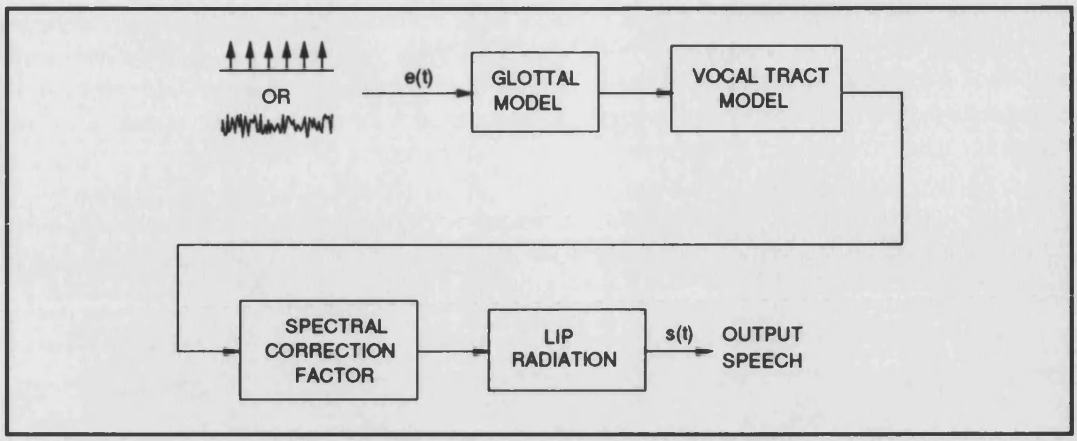

Figure 3.1 *The Linear Speech Production Model.*

The model representing the human glottis is a two pole low-pass filter with an estimated cut-off frequency at 100Hz.

The vocal tract model is an all pole model consisting of a cascade of a small number of two pole resonators. Each resonance is termed a formant having a particular centre frequency and bandwidth.

An accurate model would have an infinite number of vocal tract resonances, which would raise the spectral level at lower frequencies. In telecommunications, only the lower frequencies from 20Hz to several kHz are of interest, since all the significant energy is within these frequencies, and this shaping can be performed by a higher pole correction factor, which represents the lower frequency effects of the higher poles. It was noted by Rabiner [39] that with digital analysis/synthesis of speech that this correction term can conveniently be eliminated.

The final block is the lip radiation model where volume velocity is transformed into an acoustic pressure wave some distance away from the lips (representing the speech waveform $s(t)$).

The detailed derivation of this model is covered by Flanagan [13] along with a number of carefully conducted experiments that substantiate this model. The important results are summarised below.

This model can be described in z-transform notation

$$S(z) = E(z)G(z)V(z)L(z) \tag{3.1}$$

where

$$S(z) \Leftrightarrow s(nT) = s(t)\,|_{t=nT} \tag{3.2}$$

defines the relationship between the continuous waveform $s(t)$, the sampled waveform $s(nT)$ given by sampling $s(t)$ every T, and the z-transform $S(z)$. For the purposes of this description, a normalized sampling interval is assumed, $T = 1$, so that $s(n)$ describes the sampled $s(t)$ and similarly for other variables.

The glottal shaping model $G(z)$ is of the form

$$G(z) = \frac{1}{(1 - e^{-cT} z^{-1})^2} \tag{3.3}$$

and the lip radiation model $L(z)$ is of the form

$$L(z) = 1 - z^{-1} \tag{3.4}$$

these are generalised assumptions that do not necessarily predict the behaviour of a particular speech sample.

The all-pole vocal tract model $V(z)$, consisting of K formants, is described by

$$V(z) = \frac{1}{\prod_{i=1}^{K} \left[1 - 2e^{-c_i T} \cos(b_i T) z^{-1} + e^{-2c_i T} z^{-2}\right]} \tag{3.5}$$

where the i^{th} formant frequency and bandwidth are computed from $F_i = b_i/2\pi$ and $B_i = c_i/\pi$

As described, the model can only be excited by a periodic pulse train or white noise, and the vocal tract model only allows a number of fixed formant frequencies and bandwidths. Thus only steady state vowel or fricative sounds are defined. To represent the time varying nature of speech the input and filter parameters must be updated at regular intervals, typical rates being 50 to 100 times per second.

The combination of the glottal $G(z)$, vocal tract $V(z)$ and lip radiation $L(z)$ models is of the form

$$G(z)V(z)L(z) = \frac{(1-z^{-1})}{(1-e^{-cT}z^{-1})^2\left\{\prod_{i=1}^{K}\left[1-2e^{-c_iT}\cos(b_iT)z^{-1}+e^{-2c_iT}z^{-2}\right]\right\}} \qquad 3.6$$

where k formants are defined in the model. There is only one numerator term $(1-z^{-1})$ and it is nearly cancelled by one of the denominator terms $(1-e^{-cT}z^{-1})$ since cT is generally much less than unity. Making this simplification gives the all-pole synthesis model.

$$S(z) = E(z)\frac{1}{A(z)} \qquad 3.7$$

Where

$$A(z) = \sum_{i=0}^{M} a_i z^{-i} \approx 1/G(z)V(z)L(z) \qquad 3.8$$

noting that $M \geq 2K+1$ and $a_0 = 1$

This is termed the synthesis model since if $E(z)$ is applied to the all pole filter $1/A(z)$, the output is $S(z)$, the z-transform of the speech signal. Multiplication of both sides of equation 3.7 by $A(z)$ gives the analysis model

$$E(z) = S(z)A(z) \qquad 3.9$$

This is termed the analysis model since if the speech signal $S(z)$ is input to the inverse filter $A(z)$ (which is determined by analysing the speech waveform), the output is $E(z)$ which is the driving function to the synthesis model.

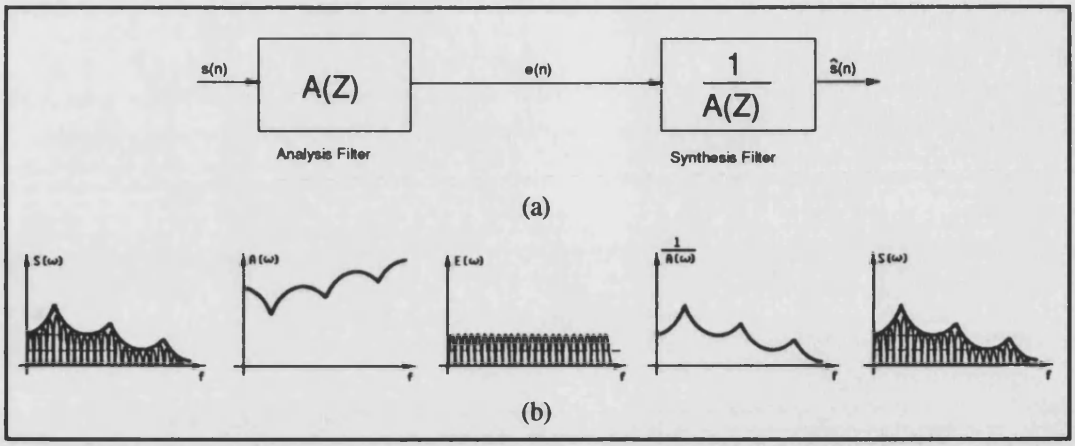

Figure 3.2 *(a) LPC analysis filtering of a speech signal, followed by synthesis filtering, (b) Frequency Spectrums (left to right) of Input Speech* $S(\omega)$*, Analysis Filter Response* $A(\omega)$*, Driving Function* $E(\omega)$*, Synthesis Filter Response* $1/A(\omega)$ *and Output Speech* $\hat{S}(\omega)$*.*

Figure 3.2a shows the passage of a speech signal $s(n)$ through the analysis model $A(z)$ giving $e(n)$, the driving function to the linear model, this is also termed the short-term analysis filter residual. This is then passed through the synthesis model $1/A(z)$ regenerating the original speech. Figure 3.2b shows the frequency spectrum of signals and filters. The residual $e(n)$ has relatively flat spectrum when filter parameters are determined which closely match the speech spectral envelope. The original speech spectral envelope is successfully recreated by the synthesis filter $1/A(z)$, the inverse to the analysis filter $A(z)$.

3.1.2 Linear Prediction Parameter Estimation

The traditional form of the Linear Prediction equation is

$$s_n = \sum_{k=1}^{p} a_k s_{n-k} + G \sum_{l=0}^{q} b_l x_{n-l} \qquad 3.10$$

where $b_0 = 1$ and G is a gain factor.

The output from a system described by this equation is a linear combination of past outputs and present and past inputs. It has already been stated in the previous section that the human speech model can be adequately described using poles only, this gives the model for human speech production

$$s_n = -\sum_{k=1}^{p} a_k s_{n-k} + e_n \qquad 3.11$$

This is illustrated in figure 3.3.

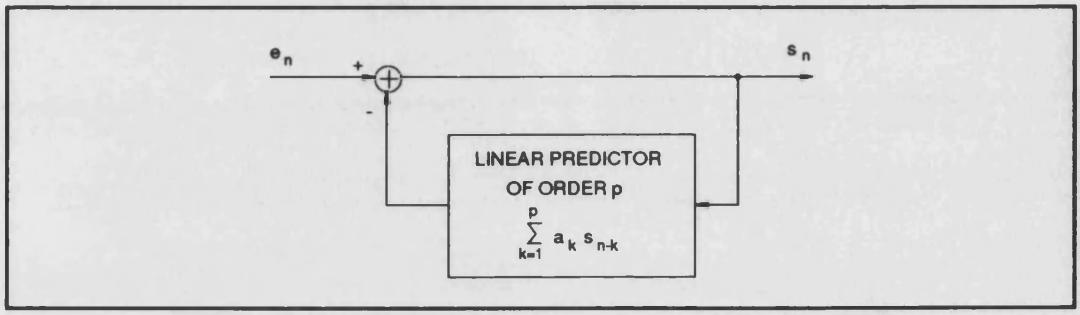

Figure 3.3 *Discrete all-pole model for human speech in time domain.*

The first problem in any LP speech encoding scheme is to determine the filter coefficients a_k. When the starting point for analysis is a speech signal s_n, the theoretical input e_n to the speech model from which this speech originated is completely unknown. Therefore the signal s_n can only be approximately predicted from a linearly weighted summation of past samples. Let this approximation of s_n be $\hat{s}_n$, where

$$\hat{s}_n = -\sum_{k=1}^{p} a_k s_{n-k} \qquad 3.12$$

The error between actual value s_n and predicted value $\hat{s}_n$ is given by

$$e_n = s_n - \hat{s}_n = s_n + \sum_{k=1}^{p} a_k s_{n-k} \qquad 3.13$$

e_n is known as the short-term prediction error, or residual. It is equal to the contribution to s_n that cannot be obtained from the linear summation of past samples. It is equivalent to the driving function of the human speech model developed in the previous section

hence it uses the same symbol (e_n).

Since e_n is the error between actual and predicted values, it makes sense to choose coefficients a_i, $i = 1,2..p$, such that e_n is somehow minimised. Historically this has been performed by minimisation of the sum of the squares of a number of error samples with respect to the coefficients a_i. The major reasons being that the resulting equations were linear, easily handled and produce excellent results with speech signals.

The total squared error E is defined, where

$$E = \sum_n e_n^2 = \sum_n \left(s_n + \sum_{k=1}^{p} a_k s_{n-k} \right)^2 \qquad 3.14$$

The range of summation in equation 3.14 has not yet been defined. However, firstly E is minimised by setting

$$\frac{\partial E}{\partial a_i} = 0, \qquad 1 \le i \le p. \qquad 3.15$$

From equations 3.14 and 3.15 the *normal equations* are derived

$$\sum_{k=1}^{p} a_k \sum_n s_{n-k} s_{n-i} = -\sum_n s_n s_{n-i}, \qquad 1 \le i \le p \qquad 3.16$$

For any sampled signal s_n, (3.16) forms a set of p equations in p unknowns which can be solved for the predictor coefficients $\{a_k, 1 \le k \le p\}$ which minimise E in 3.14.

The minimum total squared error E_p is obtained by combining 3.14 and 3.16. This gives

$$E_p = \sum_n s_n^2 + \sum_{k=1}^{p} a_k \sum_n s_n s_{n-k} \qquad 3.17$$

The range of summation over n in 3.14, 3.16 and 3.17 can now be specified. There are two cases of interest, which give rise to two different methods for estimating the parameters, and these are described in the next two subsections.

3.1.2.1 Autocorrelation Method

In this method it is assumed that the prediction error E in 3.14 is minimised over the infinite duration $-\infty < n < +\infty$. Equations 3.16 and 3.17 then reduce to

$$\sum_{k=1}^{p} a_k R(i-k) = -R(i), \qquad 1 \le i \le p \tag{3.18}$$

$$E_p = R(0) + \sum_{k=1}^{p} a_k R(k) \tag{3.19}$$

where

$$R(i) = \sum_{n=-\infty}^{\infty} s_n s_{n+i} \tag{3.20}$$

is the autocorrelation function of the signal s_n. This is an even function of i, ie. $R(-i) = R(i)$. It is also a function of subscript differences. The coefficients $R(i-k)$ form an autocorrelation matrix and equation 3.18 can be written as

$$\begin{pmatrix} R(0) & R(1) & R(2) & . & R(p-1) \\ R(1) & R(0) & R(1) & . & R(p-2) \\ R(2) & R(1) & R(0) & . & R(p-3) \\ . & . & . & . & . \\ R(p-1) & R(p-2) & R(p-3) & . & R(0) \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ . \\ a_p \end{pmatrix} = -\begin{pmatrix} R(1) \\ R(2) \\ R(3) \\ . \\ R(p) \end{pmatrix} \tag{3.21}$$

Or this can be written

$$\dot{\mathbf{R}}\mathbf{a} = -\mathbf{r} \tag{3.22}$$

This matrix, as well as being symmetrical, is also Toeplitz, meaning any diagonal has equal elements. This is exploited to give fast, efficient computational algorithms.

In practice, we are interested in the signal s_n over only a finite interval. This is achieved by multiplying the signal s_n by a window function w_n to obtain another signal s_n' that is zero outside some interval $0 \leq n \leq N-1$.

$$s_n' = \begin{cases} s_n w_n & 0 \leq n \leq N-1 \\ 0 & otherwise \end{cases} \qquad 3.23$$

The autocorrelation function is then given by

$$R(i) = \sum_{n=0}^{N-1-i} s_n' s_{n+i}', \qquad i \geq 1 \qquad 3.24$$

The shape of the window function w_n is of importance and is dependant upon the type of signal to be analysed. For signals that vary relatively quickly, such as speech sounds, they must be suitable windowed, by a window function such as a Hamming, before they can be considered quasi-stationary for LP analysis. Usually, the successive windows overlap.

3.1.2.2 Covariance Method

In contrast to the autocorrelation method, the prediction error E in 3.14 is minimised over a finite interval, say, $0 \leq n \leq N-1$. Equations 3.16 and 3.17 then reduce to

$$\sum_{k=1}^{p} a_k \psi_{ki} = -\psi_{0i}, \qquad 1 \leq i \leq p \qquad 3.25$$

$$E_p = \psi_{00} + \sum_{k=1}^{p} a_k \psi_{0k} \qquad 3.26$$

where

$$\psi_{ik} = \sum_{n=0}^{N-1} s_{n-i} s_{n-k} \qquad 3.27$$

is the covariance of the signal s_n in the given interval. The coefficients ψ_{ik} in equation 3.25 form a covariance matrix which is also symmetrical (ie $\psi_{ik} = \psi_{ki}$), however unlike

the autocorrelation matrix, the terms in each diagonal are not equal. This can be seen from

$$\Psi_{i+1,k+1} = \Psi_{ik} + s_{-i-1}s_{-k-1} - s_{N-1-i}s_{N-1-k} \qquad 3.28$$

Also apparent from 3.28 is that the values of s_n must be known for $-p \le n \le N-1$. This is a total of $p+N$ samples. The covariance method reduces towards the autocorrelation method as the interval over which n varies tends to infinity.

3.1.2.3 Computation of Predictor Parameters

In both the autocorrelation and covariance method, the predictor coefficients a_k, $1 \le k \le p$, can be computed by solving a set of p equations in p unknowns. Standard mathematical methods, such as gaussian elimination or Crout reduction can be used. These methods require $p^3/3$ operations (multiplications or divisions) and p^2 storage locations. Redundances exist in the covariance matrix and it can be more efficiently solved using Cholesky decomposition, which requires about half the computation $p^3/6$ and about half the storage $p^2/2$ of the general methods. Further redundancies exist in the autocorrelation matrix as it is Toeplitz. A very efficient way of obtaining the solution to 3.22 is by Durbin's recursive method, which is specified as follows:

$$E_0 = R(0) \qquad 3.29a$$

$$k_i = -\left[R(i) + \sum_{j=1}^{i-1} a_j^{(i-1)} R(i-j)\right] / E_{i-1} \qquad 3.29b$$

$$a_i^{(i)} = k_i \qquad 3.29c$$

$$a_j^{(i)} = a_j^{(i-1)} + k_i a_{i-j}^{(i-1)}, \qquad 1 \le j \le i-1 \qquad 3.29d$$

$$E_i = (1-k_i^2)E_{i-1} \qquad 3.29e$$

Equations 3.29a to 3.29e are solved recursively for $i = 1, 2, \ldots, p$. The final solution is given by

$$a_j = a_j^{(p)}, \qquad 1 \le j \le p \tag{3.29f}$$

A by-product of Durbin's solution is the computation of the minimum total error E_i at each step. This gives the gain required for the synthesis model.

$$G^2 = E(p) \tag{3.30}$$

Chandra and Lin [7] have experimentally compared the autocorrelation and covariance methods of linear prediction in representing a voiced waveform. From experiments, where the analysis frame was significantly greater than a pitch period (2 of 3 times) as in most LP coding schemes, they found the performance of both formulations in representing the speech waveform was more or less the same. However, only the autocorrelation method guarantees the stability of synthesis filters.

3.1.3 Lattice Formulations

The intermediate quantities k_i, $1 \le i \le p$, from Durbin's recursive method, are known as reflection coefficients or partial correlation coefficients and play a major part in LP speech coding. They are equivalent to the predictor coefficients a_i, and can be derived from each other by the following recursive relations.

K's to a's

$$a_i^{(i)} = k_i \tag{3.31a}$$

$$a_j^{(i)} = a_j^{(i-1)} + k_i a_{i-j}^{(i-1)} \qquad \begin{matrix} i = 1, \ldots, p \\ j = 1, \ldots, i-1 \end{matrix} \tag{3.31b}$$

a's to k's

$$k_i = a_i^{(i)} \tag{3.32a}$$

$$a_j^{(i-1)} = \frac{a_j^{(i)} - a_i^{(i)} a_{i-j}^{(i)}}{1 - k_i^2} \qquad \begin{matrix} i = p, \ldots, 1 \\ j = 1, \ldots, i-1 \end{matrix} \tag{3.32b}$$

For a stable synthesis filter, ie an LP filter with all the poles inside the unit circle

$$-1 < k_i < 1 \qquad i = 1, \ldots, p \tag{3.33}$$

Thus by ensuring that all the reflection coefficients obey this condition, even after quantisation, the stability of the synthesis filter can be guaranteed. Alternatively there in no equivalent condition to ensure stability with the predictor coefficients a_i. For this reason, historically, the synthesis model has been represented by reflection coefficients.

In order to implement the synthesis filter $1/A(z)$, it is not necessary to convert reflection coefficients to predictor coefficients and then use a direct form implementation of the filter. Instead, it is possible to implement the filter in lattice form using the reflection coefficients k_i directly. Figure 3.4 shows the lattice implementation of the LP synthesis filter.

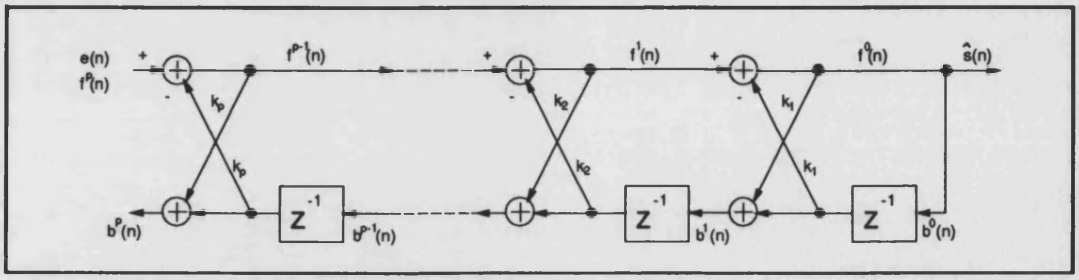

Figure 3.4 *Lattice synthesis filter with excitation $e(n)$ and reflection coefficients, k_i, giving output $\hat{s}(n)$.*

The equations implementing the lattice filter are

$$f^{(i-1)}(n) = f^{(i)}(n) - k_i b^{(i-1)}(n-1) \qquad 3.34a$$

$$b^{(i)}(n) = k_i f^{(i-1)}(n) + b^{(i-1)}(n-1) \qquad 3.34b$$

with

$$f^{(p)}(n) = e(n) \qquad 3.35$$

and $e(n)$ is the excitation, from the original short-term linear prediction analysis. The superscript indicates the stage in the lattice filter, while the argument is the time index. The output is given by

$$s(n) = f^{(0)}(n) \qquad 3.36$$

The above equations have introduced the quantities $f^{(i)}(n)$ and $b^{(i)}(n)$, which are called the forward and backward error respectively. If equations 3.34a and 3.34b are written in the form

$$f^{(i)}(n) = f^{(i-1)}(n) + k_i b^{(i-1)}(n-1) \qquad 3.37a$$

$$b^{(i)}(n) = b^{(i-1)}(n-1) + k_i f^{(i-1)}(n) \qquad 3.37b$$

they represent the analysis filter $A(z)$. When the input to the above filter is the speech signal $s(n)$, the output is the residual error signal $e(n)$. This filter is implemented as illustrated in figure 3.5 with input

$$f^{(0)}(n) = b^{(0)}(n) = s(n) \qquad 3.38$$

and output

$$e(n) = f^{(p)}(n) \qquad 3.39$$

Thus both analysis and synthesis filters can be implemented in lattice form!

Durbin's method is effective for calculating both forms of LP parameters, however if only reflection coefficients are required, the filter coefficients a_i must be computed as

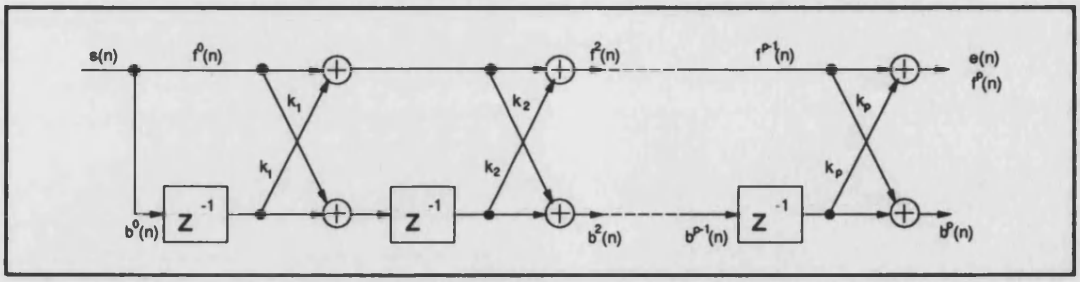

Figure 3.5 *Lattice analysis filter. The input is the speech signal $s(n)$. The output is the residual error $e(n)$.*

intermediate quantities. These have a large dynamic range, and calculation could become troublesome in fixed point arithmetic, such as with some digital signal processor devices. This problem could be eased by using the Leroux-Guegen analysis method [30] which is equivalent to the Schur Recursion used in the GSM Pan-European speech coder.

3.1.4 Transmission of Filter Parameters

Linear predictive analysis produces a set of p real-valued predictor coefficients a_k, which represent an optimal estimate to the spectrum of the original speech using p poles. These predictor coefficients a_k have an unacceptably large range for efficient quantisation, this also causes problems with fixed point implementation of filters. Reflection coefficients offer some improvement, but again also have their own associated quantisation problem. Better methods include the log-area ratio or the line spectrum pair which are described in the next two sections.

3.1.4.1 Log-Area Ratios

The stability of a lattice synthesis filter is guaranteed by ensuring that all the dequantised reflection coefficients have magnitude less than unity. However, when they have values close to the boundaries -1 or +1, the synthesis filter is very sensitive to reflection coefficients quantisation errors. Transforming to log-area-ratios (LARs) before

transmission equalises the sensitivity to quantisation errors for all values [53], such that uniform quantisation of the LARs corresponds to non uniform quantisation of the reflection coefficients.

The log-area ratios are defined by

$$LAR_i = \log\frac{1+K_i}{1-K_i} \qquad i = 1,\ldots,p \qquad 3.40$$

Conversely, the reflection coefficients can be recovered from the LAR by

$$K_i = \frac{e^{LAR_i}-1}{e^{LAR_i}+1} \qquad i = 1,\ldots,p \qquad 3.41$$

When the transmitted parameter is quantised with only several bits, the use of logarithm and exponential functions, which are impractical for real time systems, can easily be avoided using stepwise linear approximations. This is discussed with reference to the GSM speech coder in section 4.3.2.

3.1.4.2 Line Spectrum Pairs

Historically, the log-area ratios were most widely used for transmission of filter parameters, however another representation of LP filter parameters that is gaining in popularity is the line spectrum pair (LSP). This gives a reduction of about 25% in bit rate than with reflection coefficients, whilst maintaining an equivalent quality [22]. One useful property of the LSP is that an error in one line-spectrum affects the all-pole spectrum near that frequency. Thus LSPs may be quantised in accordance with auditory perception, enabling coarser quantisation of the high-frequency components of the speech spectral envelope.

The LSP is defined by the decomposition of the impulse response of the LP analysis filter into even and odd functions. In terms of predictor coefficients, the transfer function of the p^{th} order LP analysis filter is

$$A(z) = 1 - a_1 z^{-1} - a_2 z^{-2} - \ldots - a_p z^{-n} \qquad 3.42$$

$A(z)$ is decomposed into two transfer functions, one having an even symmetry and the other having an odd symmetry. This is achieved by taking a difference and sum between $A(z)$ and its conjugate function (ie the transfer function of the filter whose impulse response is a mirror image of the LP analysis filter). This gives the difference filter

$$P_{n+1}(z) = A_n(z) - z^{-(n+1)} A_n(z^{-1}) \qquad 3.43$$

and the sum filter

$$Q_{n+1}(z) = A_n(z) + z^{-(n+1)} A_n(z^{-1}) \qquad 3.44$$

The LP analysis filter reconstructed from these two filters is

$$A(z) = \frac{1}{2}[P_{n+1}(z) + Q_{n+1}(z)] \qquad 3.45$$

Roots of the analysis filter are located inside the unit circle of the z-plane, whereas roots of both the sum and difference filters are located on the unit circle. The LSP method does require finding the roots of both the P and Q polynomials. However since they lie on the unit circle, and because they are quantised with a low number of bits, this can be accomplished quite efficiently. The low bit rate approach of the LSP is gaining wide acceptance and has been incorporated into the U.S. Federal Standard 1016 CELP coder [6].

3.1.5 Error Weighting Filters for Speech Coders

Section 2.5 introduced the subject of masking, where the perception of one sound is obscured by the presence of another. In designing a speech coder, rather than minimising output noise power, it would be far better to minimise its subjective loudness. Noise is heard less at frequencies of strong speech energy, ie at the low frequency formants. In

modern analysis-by-synthesis linear predictive speech coding, the perceptual error is minimised between input and synthetic speech. A filter is required that attenuates those frequencies where the error is perceptually less important and enhances those frequencies where the error is perceptually more important, ie between the formant regions. The resulting mean squared error is minimised.

The determination of possible error filters for LP coding was originally investigated by Atal and Schroeder [1], and a usual choice for it is

$$W(z) = \frac{1 - \sum_{k=1}^{p} a_k z^{-k}}{1 - \sum_{k=1}^{p} a_k b^k z^{-k}} \qquad 3.46$$

b is an additional parameter introduced to increase the bandwidths of the formants, it is a number between 0 and 1 and a typical value is 0.8. An example of the envelope of an error weighted noise spectrum together with the corresponding speech spectrum is shown in figure 3.6.

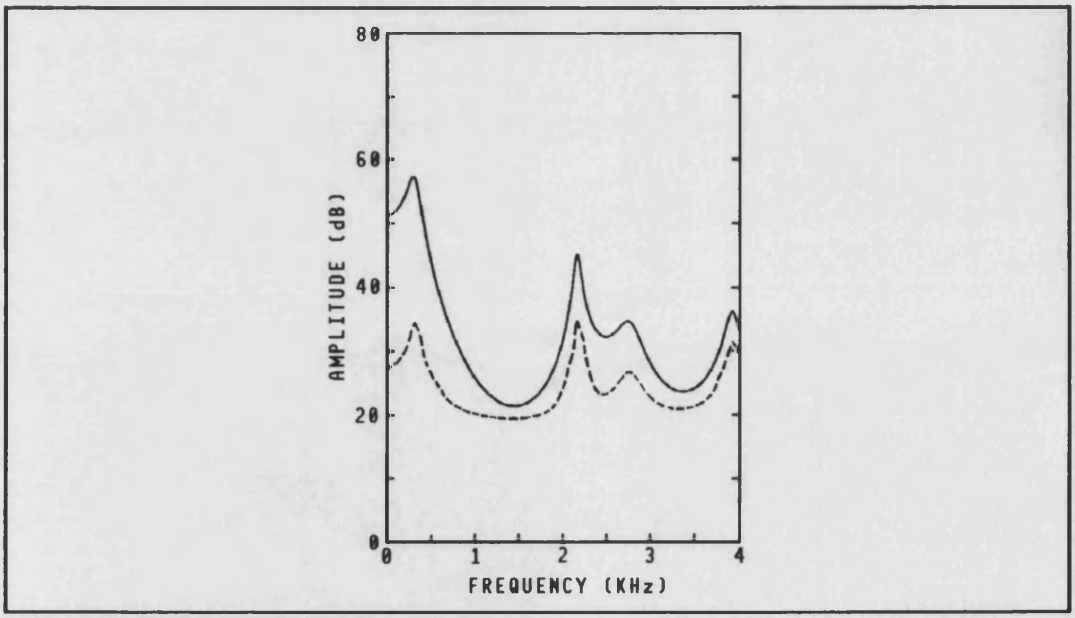

Figure 3.6 *An example showing the envelope of the output noise spectrum (dotted curve) shaped to reduce perceived distortion and the corresponding speech spectrum (solid curve), after [1].*

3.2 Long-Term Predictors (LTPs)

With voiced speech the LPC prediction residual is highly correlated between adjacent pitch periods. This periodicity can be modelled by long-term or pitch prediction and this is widespread in modern speech coding. This section covers long-term prediction as used within this thesis. Subsections 3.2.1 and 3.2.2 describe the parameter optimisation in both analysis and synthesis modes respectively. Section 3.2.3 describes the extension of the LTP delay range by the cyclic repeating of codebook elements. The final section, 3.2.4, describes long-term prediction with non-integer sample delay values.

The general form of a pitch predictor of order q is given by

$$P_e(z) = \sum_{k=0}^{q-1} \gamma(k) z^{-(d+k)} \qquad 3.47$$

where the delay d represents the pitch period (or possibly an integral number of pitch periods) in samples and $\gamma(k)$ are the pitch predictor coefficients. The delay d would be fixed for steady vowels and random for non-periodic, unvoiced signals. The predictor parameters d and $\gamma(k)$ are normally determined between 100 and 200 times per second. The number of coefficients q typically varies from 1 to 3, although the pitch predictor has often been applied in single tap form.

$$P_e(z) = \gamma z^{-d} \qquad 3.48$$

Figure 3.7a depicts the LTP operating in analysis mode, where the LTP filters out pitch periodicity, and figure 3.7b shows operation in synthesis mode, where the LTP recreates pitch periodicity. Optimisation of predictor coefficient can be performed in either mode, with analysis mode requiring significantly less processor power and is generally used in low-complexity speech coders, whereas synthesis mode is used in high complexity analysis-by-synthesis coders. In analysis mode, LTP coefficients are optimised such that the squared error between the original LPC residual signal and its predicted

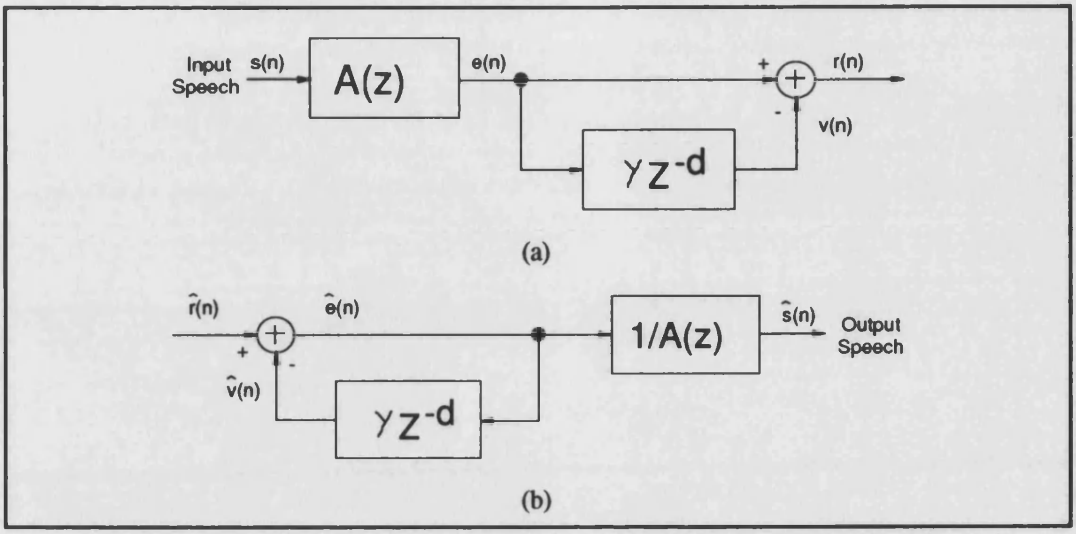

Figure 3.7 *Short and Long-Term Predictors, (a) Analysis Mode, (b) Synthesis Mode.*

counterpart is minimised. In synthesis mode, LTP coefficients and the short-term predictor coefficients are optimised such that the squared error between original and synthetic speech is minimised. In both modes, for a given delay d, coefficients $\gamma(k)$ can be found by solving a set of linear simultaneous equations [40]. Optimum delay d is found by performing an exhaustive search over its allowed range, which for the purposes of this thesis is 2.5-20ms. This corresponds to a range of fundamental frequencies of 50-400Hz for human speech.

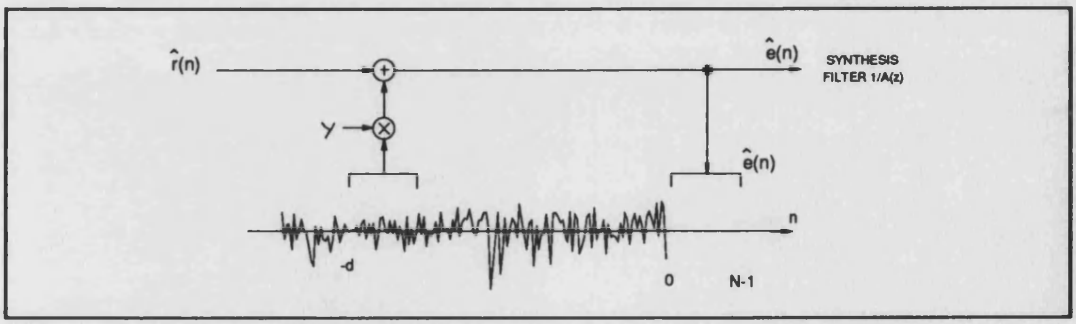

Figure 3.8 *An Alternative Representation of the LTP, Showing the Summation of Waveform Extracts (Synthesis Mode).*

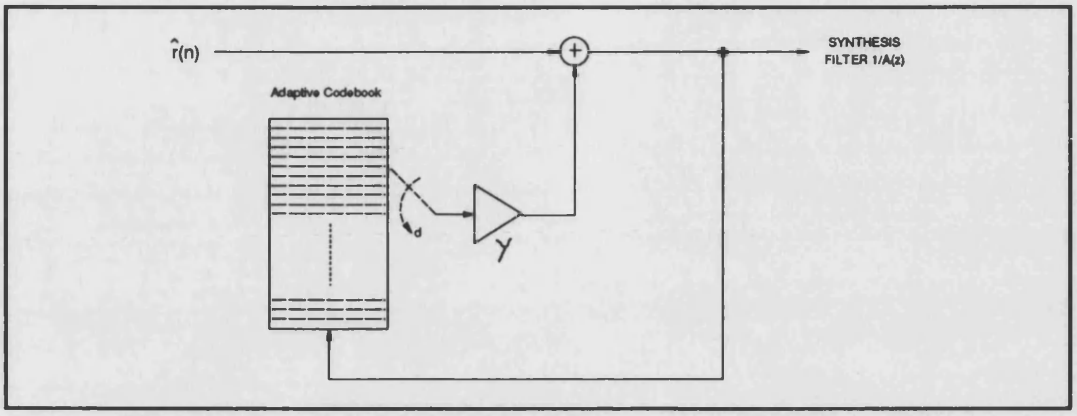

Figure 3.9 *The Adaptive Codebook Interpretation of the LTP (Synthesis Mode).*

LTP parameters are determined over successive, non overlapping subframes of length 5 to 10ms. There are usually several LTP subframes to every short-term prediction frame. A LTP must be capable of modelling pitch periods up to 20ms, requiring it to maintain a history of its own previous output for this length of time. An alternative representation of the LTP synthesis is shown in figure 3.8. This illustrates the LTP history buffer storing a waveform segment from which an excitation vector is extracted. This is then amplified before being summed with the decoded short and long-term residual $\hat{r}(n)$ to give the LTP output. This output subframe is also used to update the LTP history ready for processing the next subframe. Since the LTP extracts excitation vectors from its own output history, it is often termed an adaptive codebook, where adjacent codebook entries overlap for all but one sample. Figure 3.9 illustrates this adaptive codebook interpretation of the LTP.

A single tap LTP is very effective when the pitch period is close to an integral number of sample periods, however its performance is limited otherwise. A multitap predictor acts as an interpolating filter and allows LTP operation over non-integer pitch periods. The extent to which LTPs remove redundancies is determined by measuring the energy of the resulting residual. The prediction gain is the ratio of the average energy at the

input of the LTP to the average energy of the prediction residual. Prediction gains are higher with multitap pitch predictors, however the associated mathematics is considerably more complex and does not always guarantee stable filters [40]. A more recent technique is to use single tap LTPs which operate with non-integer delays, which has been shown to give higher prediction gains than 3-tap LTPs [25]. In addition there is a bit rate saving. LTPs used in this thesis were all single tap and those in chapters 6 and 7 utilise non-integer pitch periods.

3.2.1 LTP Analysis

In analysis mode, depicted in figure 3.7a, the short-term residual signal $e(n)$ is approximated by the LTP output signal $v(n)$, which is a function of both delay d and LTP gain coefficient γ. The difference between these two signals is the short and long term prediction error $r(n)$. Output from the LTP is given by

$$v(n) = \gamma e(n-d) \tag{3.49}$$

Thus the short and long-term residual error

$$r(n) = e(n) - \gamma e(n-d) \tag{3.50}$$

The squared prediction error for a subframe of N samples is then given by

$$E(d,\gamma) = \sum_{k=0}^{N-1} r^2(n) \tag{3.51}$$

Thus

$$E(d,\gamma) = \sum_{k=0}^{N-1} [e(n) - \gamma e(n-d)]^2 \tag{3.52}$$

For a given delay d, to determine the optimum LTP gain, minimisation of E with respect γ gives

$$\gamma = \frac{\sum_{k=0}^{N-1} e(k)e(k-d)}{\sum_{k=0}^{N-1} e^2(k-d)} \qquad 3.53$$

Substituting this optimum gain into equation 3.52, leads to the error function

$$E(d) = \sum_{k=0}^{N-1} e^2(k) - E'(d) \qquad 3.54$$

where

$$E'(d) = \frac{\left[\sum_{k=0}^{N-1} e(k)e(k-d)\right]^2}{\sum_{k=0}^{N-1} e^2(k-d)} \qquad 3.55$$

This function $E'(d)$ is calculated for all possible delays d, and the maximum corresponds to the optimum delay.

In a practical LTP implementation, the LTP adaptive codebook is a buffer of length L samples

$$\{v_{-i}\} = \{v_{-L}, v_{-L+1}, v_{-L+2}, \ldots, v_{-1}\} \qquad 3.56$$

The search through the history of $e(n)$ for the optimum excitation, is in practice a search through this buffer. For convenience, after delay and gain optimisation, the latest synthesis filter excitation vector, is placed in the notional continuation of this codebook buffer, ie in elements $[v_0..v_N]$. The adaption of the codebook ready for the next subframe is then simply the left shifting of this buffer by the analysis subframe length L.

$$v(i) = v(i+N) \qquad -L \le i < 0 \qquad 3.57$$

Figure 3.10 illustrates operation of a single tap LTP in analysis mode. The top trace is the input speech waveform ("seven"). This starts unvoiced (*s*), voiced fricates (*v*), a combination of voicing and unvoicing, and finally finishes voiced (*en*). The words three portions have widely differing characteristics. The middle trace is the short-term

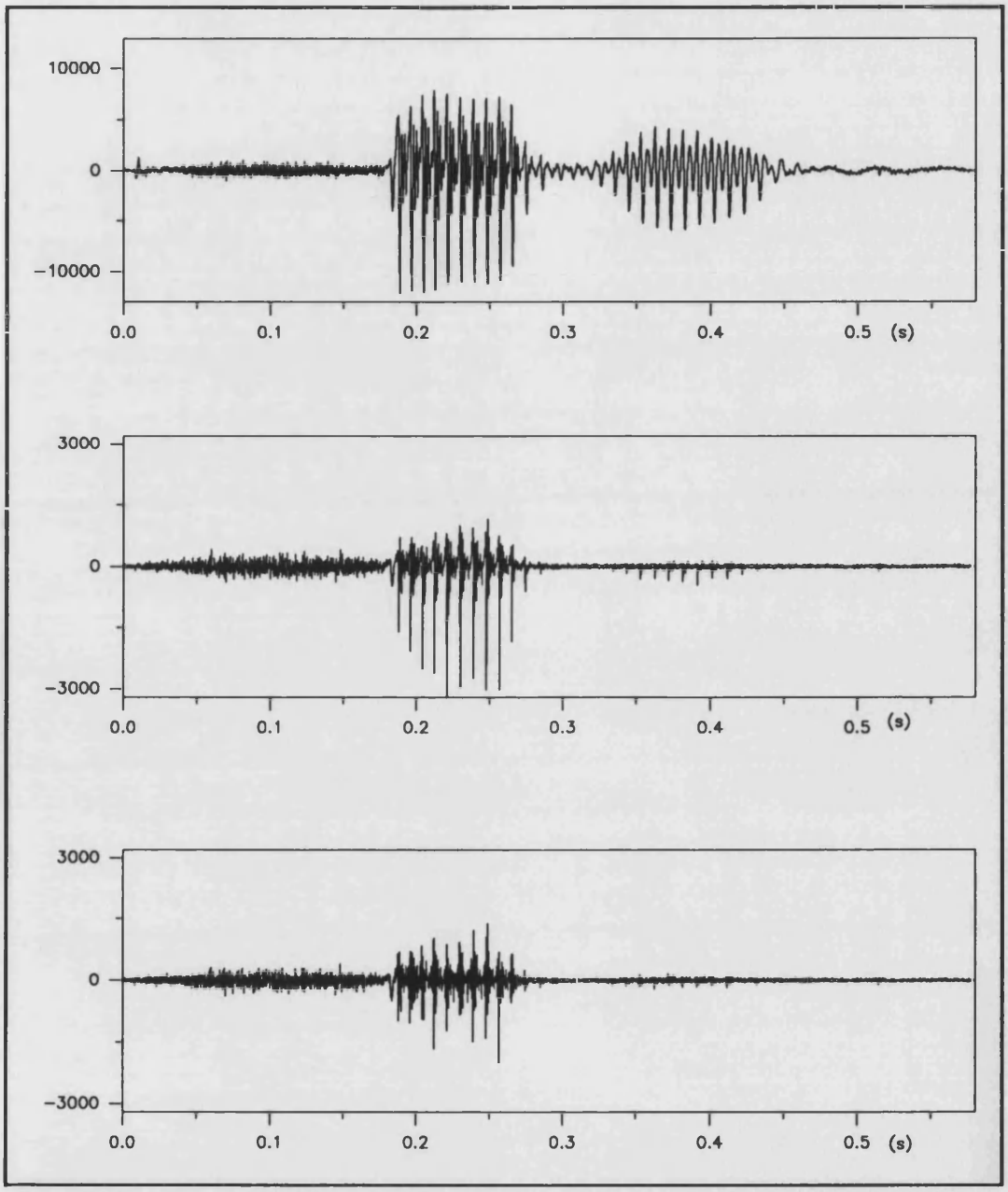

Figure 3.10 *Operation of a Single Tap LTP in Analysis Mode, top: input speech ("Seven"), middle: waveform after short-term analysis filtering, bottom: waveform after long-term analysis filtering. (Note Waveforms (b) and (c) are shown amplified by 13 dB).*

analysis filter residual and the bottom trace is the signal after long-term analysis filtering (middle and bottom traces have are shown amplified by 13dB relative to input speech). This bottom waveform has lower amplitude and shows considerably less pitch structure than the middle trace.

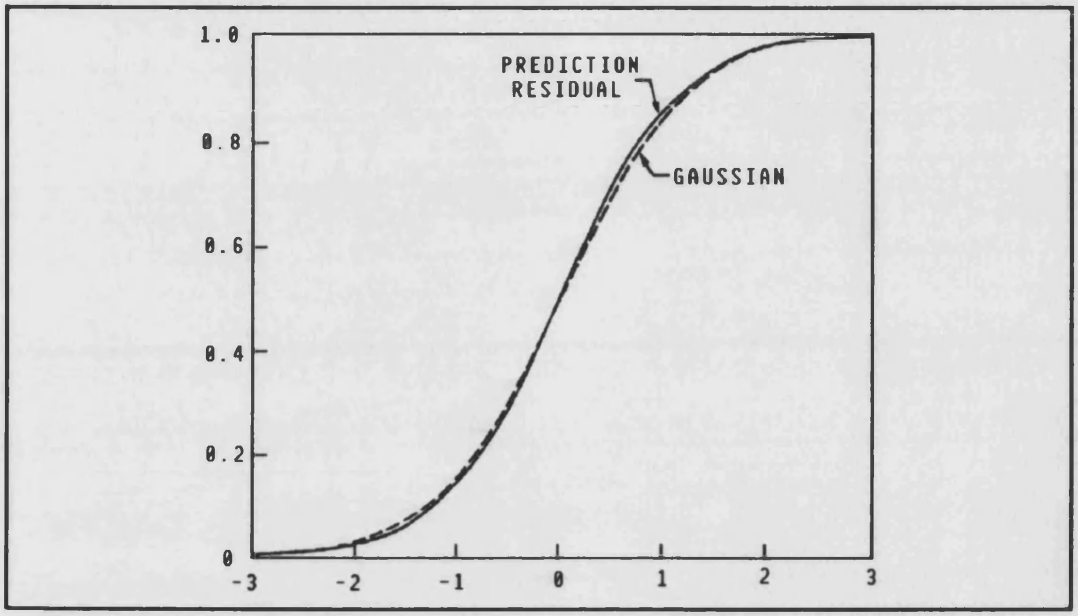

Figure 3.11 *First-order cumulative probability distribution function for the prediction residual samples (solid curve). The corresponding Gaussian distribution function with the same mean and variance is shown by the dashed curve, after [2].*

Figure 3.11 shows a plot of the first-order cumulative amplitude distribution function for the short and long-term prediction residual and compares it with the corresponding Gaussian distribution function with the same mean and variance. This Gaussian assumption is valid almost everywhere [2], except for stop bursts of unvoiced stop consonants and for a few pitch periods during the transition from unvoiced or silence regions to voiced speech.

3.2.2 LTP Synthesis

In high complexity speech coders, the LTP operates in synthesis mode, depicted in figure 3.7b. The determination of optimum gain and delay is now performed with synthetic speech rather than with residual error. Thus the equations from the previous

section are modified to include convolution with the synthesis filter impulse response. The synthesis filter $1/A(z)$ input is given by

$$\hat{e}(n) = \hat{r}(n) + \gamma \hat{e}(n-d) \qquad 3.58$$

Where γ is the predictor gain, d is the predictor delay and $\hat{r}(n)$ is the decoded short and long-term residual, this could be derived from a codebook vector, multipulses, residual pulses etc.

The synthetic speech signal $\hat{s}(n)$ is given by

$$\hat{s}(n) = \hat{r}(n) * h(n) + \gamma \hat{e}(n-d) * h(n) + \hat{m}(n) \qquad 3.59$$

where * denotes convolution and is a memoryless process, $h(n)$ is the impulse response of the all-pole synthesis filter $1/A(z)$ at the n^{th} sampling instant. The contribution $\hat{m}(n)$ is that due to the memory of the all-pole filter. It is calculated by maintaining the initial filter state after synthesizing the previous speech subframe and exciting the filter with the zero vector. This contribution exists in every speech frame. Let

$$\tilde{s}(n) = s(n) - \hat{m}(n) \qquad 3.60$$

$\tilde{s}(n)$ is termed the target input speech vector.

To determine the optimum prediction parameters, it is assumed that the decoded excitation sequence is zero and we minimise the total squared error E to compute the predictor gain γ for a particular delay d. The mean squared error between target vector and synthetic speech vector is given by

$$E(d,\gamma) = \sum_{n=0}^{N-1} \tilde{s}^2(n) - \gamma \sum_{n=0}^{N-1} \tilde{s}(n)\,[\hat{e}(n-d) * h(n)] \qquad 3.61$$

For a given delay d, minimisation of E with respect to γ gives

$$\gamma = \frac{\sum_{n=0}^{N-1} \tilde{s}(n)\,[\hat{e}(n-d) * h(n)]}{\sum_{n=0}^{N-1} [\hat{e}(n-d) * h(n)]^2} \qquad 3.62$$

and

$$E(d) = \sum_{n=0}^{N-1} \tilde{s}^2(n) - E'(d) \qquad 3.63$$

where

$$E'(d) = \frac{\left(\sum_{n=0}^{N-1} \tilde{s}(n)\,[\hat{e}(n-d) * h(n)] \right)^2}{\sum_{n=0}^{N-1} [\hat{e}(n-d) * h(n)]^2} \qquad 3.64$$

The optimum predictor delay d is that which maximises $E'(d)$ in equation 3.64. The predictor gain γ is then calculated from equation 3.62.

Figure 3.12 illustrates operation of a single tap LTP in synthesis mode. The top trace is the decoded short and long-term analysis residual of the utterance "Seven". In this case it has been reconstructed after transmission as residual pulses. It shows negligible periodicity. The middle trace shows the waveform after long-term synthesis filtering and the periodicity has been notably re-created. The bottom trace shows the synthetic speech produced after short-term synthesis filtering. Top and middle traces are shown amplified by 13dB relative to bottom trace.

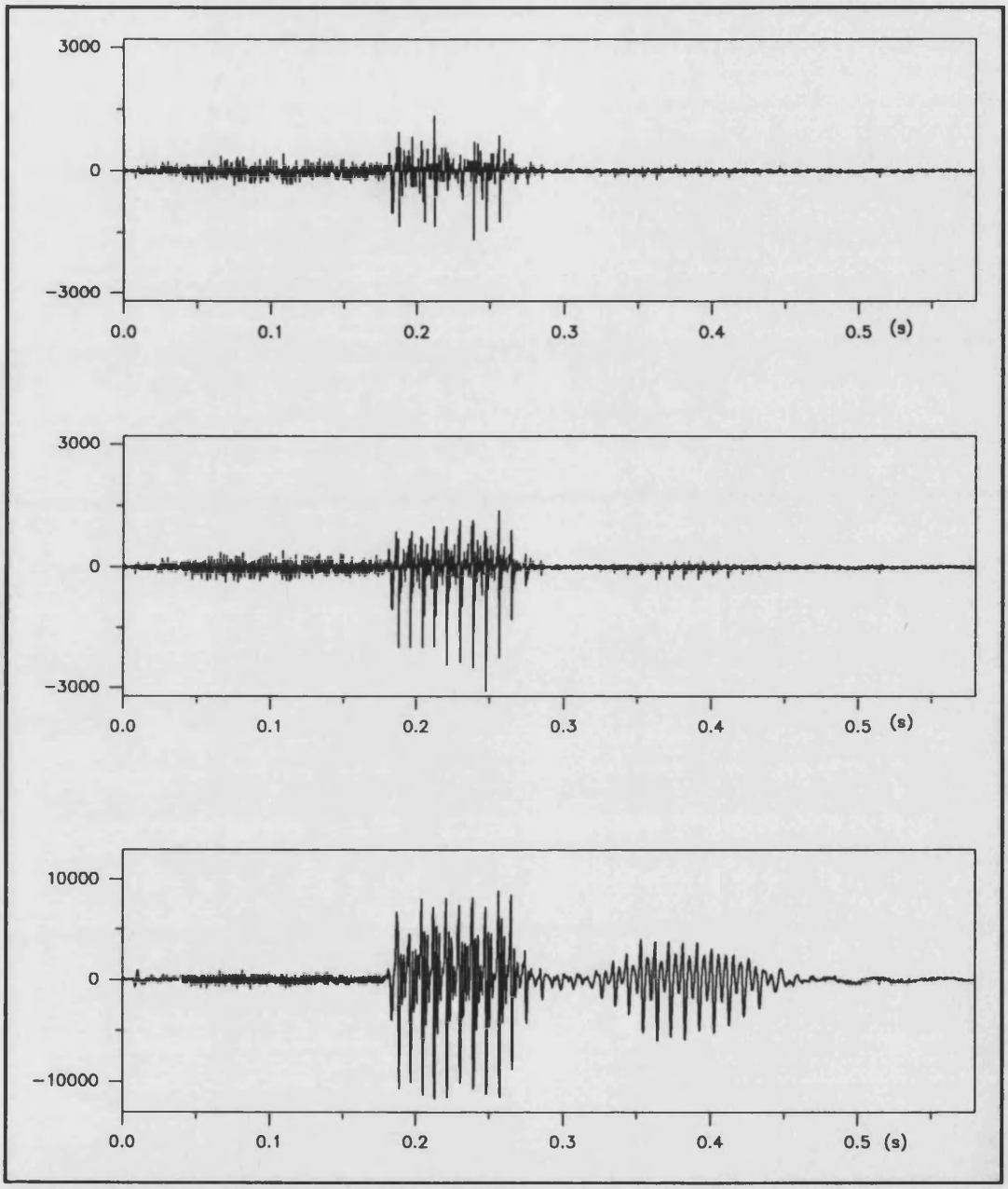

Figure 3.12 *Operation of a Single Tap LTP in Synthesis Mode, top: Short and Long-Term Residual, middle: Waveform after Long-Term Synthesis Filtering, bottom: Synthetic Speech ("seven"). (Note Waveforms (a) and (b) are shown amplified by 13 dB).*

3.2.3 Extending the LTP Range

Human speech fundamental frequencies have overall range from 50 to 400Hz. In current linear predictive speech coding, the residual signal is normally encoded over subframes

of either 40 or 60 samples, corresponding to 5 or 7.5ms respectively. Within this thesis, LTPs operate on subframes of 40 samples and with the simpler versions of chapters 4 and 5, the minimum possible delay is equal to this subframe length. With the minimum delay 40, candidate excitation vector $[v_{-40}..v_{-1}]$ is extracted from the adaptive codebook, whereas with maximum delay L, vector $[v_{-L}..v_{-L+40}]$ is extracted.

A minimum delay equal to the LTP subframe length is a serious limitation in modelling high fundamental frequency speakers. Often, the LTP has to settle for modeling a multiple of the pitch. It is far better to modify the LTP delay range to cover the higher fundamental frequencies, and in more modern speech coders, LTPs generally operate over delay ranges of 2.5-20ms. Assuming that the subframe length remains at 40 samples, delays d of 20 to 39 are obtained by cyclically repeating codebook elements from $[v_{-d}..v_{-1}]$ to form a 40 sample vector. For example, delay 20 corresponds to $[v_{-20}..v_{-1}, v_{-20}..v_{-1}]$, delay 21 corresponds to $[v_{-21}..v_{-1}, v_{-21}..v_{-3}]$ and delay 39 corresponds to $[v_{-39}..v_{-1}, v_{-39}]$.

3.2.4 Non-Integer Delays in LTPs

Using a conventional single tap LTP, the delay d is expressed as an integer number of samples at sampling rate f_s. Non-integer pitch delays are expressed as an integer number of samples at rate f_s plus a fraction of a sample l/D, where $l = 0, 1, \ldots, D-1$, (l and D are integers.) A non-integer delay l/D at the original sampling rate f_s corresponds to an integer delay at the higher sampling rate Df_s. Thus to produce non-integer delays in LTPs, the sampling rate of the LTP adaptive codebook must be interpolated to Df_s.

The interpolating process is illustrated in figure 3.13. Firstly, the input signal, in this case the LTP adaptive codebook, is up-sampled, by inserting $D-1$ zeros between each sample. This increases the sample rate, but it also introduces alias components in the signal spectrum. The next stage is low-pass filtering to remove these alias components. The next stage introduces a delay of l samples at this higher sample. The final stage is

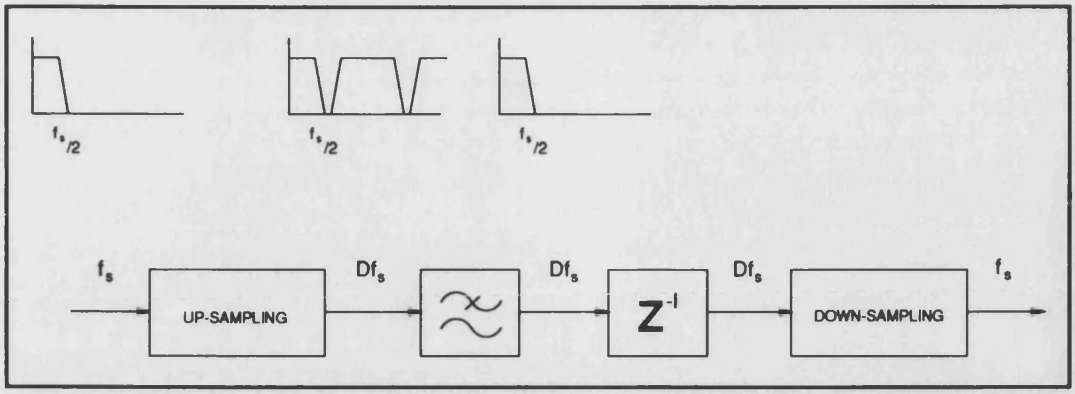

Figure 3.13 *The Interpolation Process used in Non-Integer LTPs, Above Graphs Show the Signal Frequency Spectrum.*

down-sampling to the original sampling frequency f_s, by extracting equidistant samples D samples apart. The resulting signal is the original signal delayed by non-integer delay l/D.

In a practical system, the low-pass filter will also introduce a delay. The interpolation filter $h(n), n = 0, 1, \ldots, \mathbf{N} - 1$ is chosen to be a FIR filter with exactly linear phase and a unit sample response of duration **N**. Its delay at the high sampling rate Df_s is $(\mathbf{N} - 1)/2$ samples. To compensate for this overall delay at the lower sampling rate, **N** must be chosen such that $(\mathbf{N} - 1)/2$ is an integer multiple of D, ie

$$\mathbf{N} = 2ID + 1 \qquad 3.65$$

where I is the delay at original sampling frequency f_s. For computational efficiency, the non-integer delays are implemented with a polyphase structure as described in appendix 1.2.3. This low-pass filter must attenuate all components above $f_s/2$ to prevent aliasing components after the down-sampling process.

Figure 3.14 demonstrates the interpolation process, the top trace is an extract of short-term analysis residual, each spike representing one sample, the bottom trace is

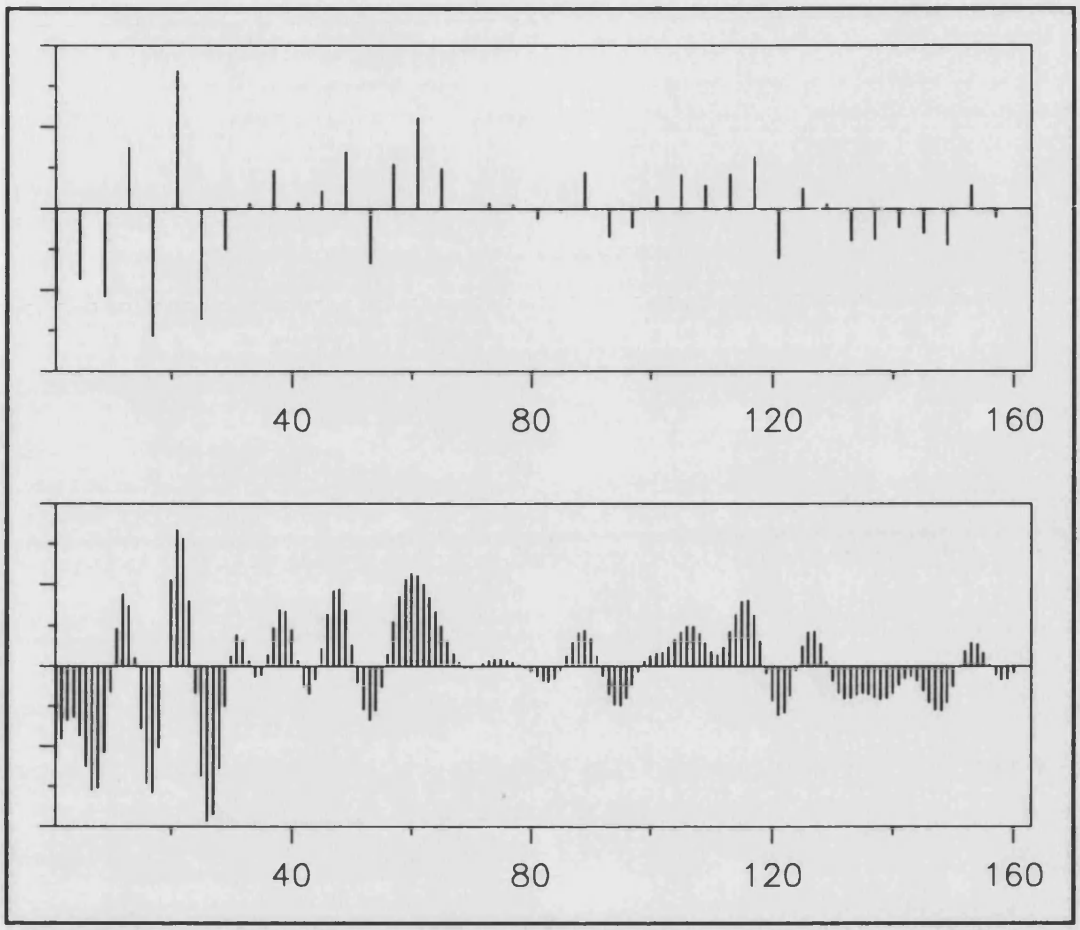

Figure 3.14 *Demonstration of Interpolation, top: Input Waveform, bottom: Interpolated Waveform.*

the same signal after up-sampling and low-pass filtering. From this interpolated signal, integer delay vectors and 1/4, 1/2 and 3/4 sample non-integer delay vectors can be extracted.

3.3 Linear Predictive Speech Coders

The previous two major sections described in detail short and long-term prediction of speech. This section introduces several practical speech coders which use both these predictions. The section will split speech coders into two types, these being "low complexity" and "high complexity", (or analysis-by-synthesis). Another distinction used in this section will be the distinction between "residual encoders" and "excitation generators" as the excitation source. It must be stressed that speech coding is a very widely researched area and these classifications although useful, are a vast over simplification.

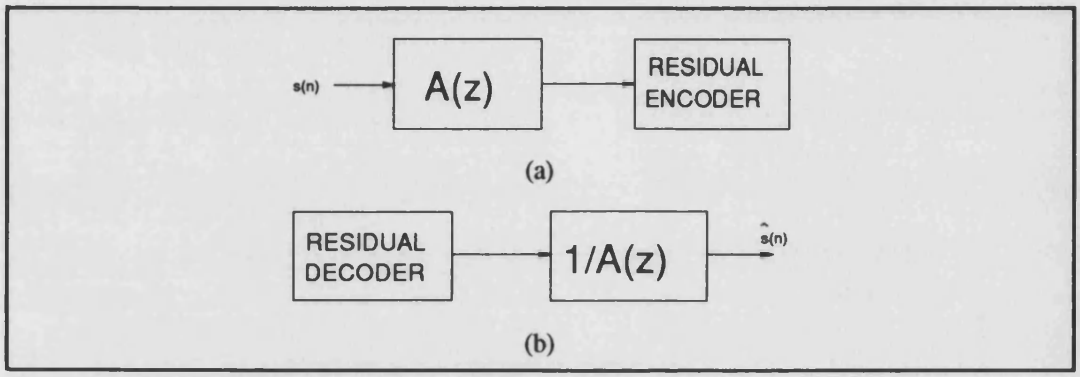

Figure 3.15 *Low Complexity LPC Speech Coder, (a) Encoder, (b) Decoder.*

The general low complexity speech encoder is depicted in figure 3.15a. Input speech is filtered by short-term analysis filter $A(z)$. The resulting residual is then directly encoded, in manner dependant upon the type of speech coder. The decoder, depicted in figure 3.15b, is the inverse to the encoder, recreating the residual from the transmitted information and inputting it to the short-term synthesis filter $1/A(z)$ giving synthetic speech. This approach is used in the source vocoder and the RELP vocoder described in sections 3.3.1 and 3.3.2 respectively. This approach is also used in the RPE coder of section 3.3.3 which can be improved with inclusion of a LTP.

The high complexity speech coders differ from their low complexity counterparts in having "excitation generators" rather than "residual encoders". The residual encoder

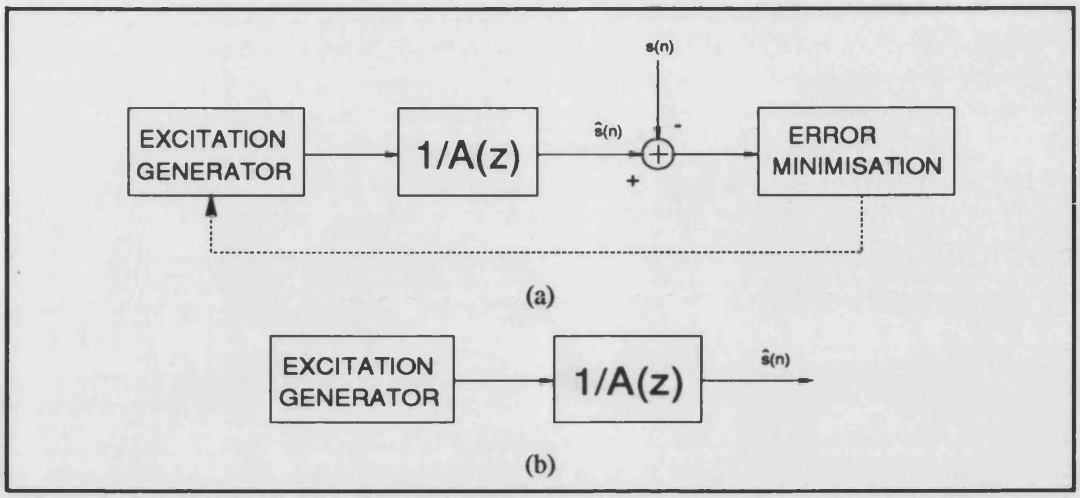

Figure 3.16 *High Complexity, Analysis-by-Synthesis, LPC Speech Coder, (a) Encoder, (b) Decoder.*

attempted to encode all the important residual information, whereas the excitation generator is capable of producing many candidate excitation sequences and optimisation consists of selecting that sequence which gives best synthetic speech. Figure 3.16 illustrates these high complexity coders, from this diagram it can be clearly seen that the encoder block diagram contains its own local decoder. Once error between input and synthetic speech has been minimised, the optimum excitation parameters are transmitted to the remote decoder. Hence these coders are termed "Analysis-by-synthesis." With the similarity between encoder and decoder in these high complexity techniques, often only the block diagram for the decoder is given and it is assumed that the encoder contains the error minimisation loop. It should be noted at this stage, that it would be possible to match excitation generator candidate sequences with the actual short-term analysis residual, however in practice this results in poor speech quality. This analysis-by-synthesis approach is used in the Multipulse, Code and Self excited vocoders described in sections 3.3.4, 3.3.5 and 3.3.6 respectively.

3.3.1 The LPC Source Vocoder

The residual encoder block of the LPC source vocoder implements the linear model of speech production directly. The short-term analysis filter residual is represented as

either a periodic pulse train or as random noise, depending upon whether it decides speech is voiced or unvoiced. This leads to very low bit rates of around 2.4 kbits/s. The decoder is illustrated in figure 3.17, showing pulse/noise generation, amplification and synthesis filtering.

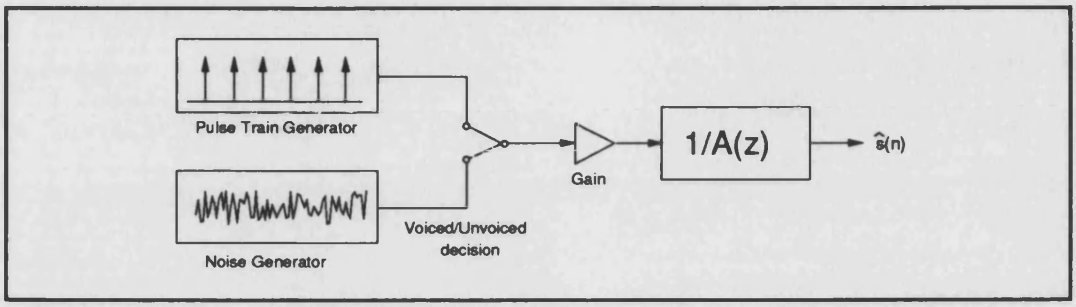

Figure 3.17 *Linear-Predictive Vocoder with a Pitch Impulse Generator for Voiced Frames and a Noise Generator for Unvoiced Frames*

A standard algorithm for speech transmission at 2400 bits/s based on this method has been developed. It is referred to as the NSA (or the DoD standard) LPC-10 algorithm [50], this uses a 10th order linear prediction model. The speech frame period is 22.5ms giving 54 bits/frame.

The output speech from such a vocoder is highly intelligible, but very mechanical sounding, especially on voiced sections. Only slight quality improvements are possible as the bit rate is increased, the limitations are due to the simplistic LPC model and the binary decision of whether the speech in any analysis frame is voiced or unvoiced. Voiced fricatives have a periodic envelope with a noise excitation and are thus poorly modelled. In addition the accurate determination of pitch in voiced speech is difficult and often unsatisfactory. The subject of pitch determination is discussed at length by Papamichalis [37] who describes 5 different algorithms. No single algorithm has been shown to perform satisfactorily in all cases. Applications for this type of speech coder are limited to those where absolute minimum transmitted bit rate is required.

3.3.2 Baseband RELP Vocoder

The baseband residually excited linear predictive (RELP) vocoder [49], illustrated in figure 3.18, exploits the relatively flat nature of the short-term analysis residual. The most important perceptual information is contained in the lower frequencies so the residual is low pass filtered and only the lower 800Hz is transmitted to the decoder. This allows the filtered signal to be uniquely described with 1/5 of the sample rate which is achieved by down sampling, ie selecting every fifth sample. This process reduces the transmission rate of the residual information from 20kbit/s to about 5kbit/s.

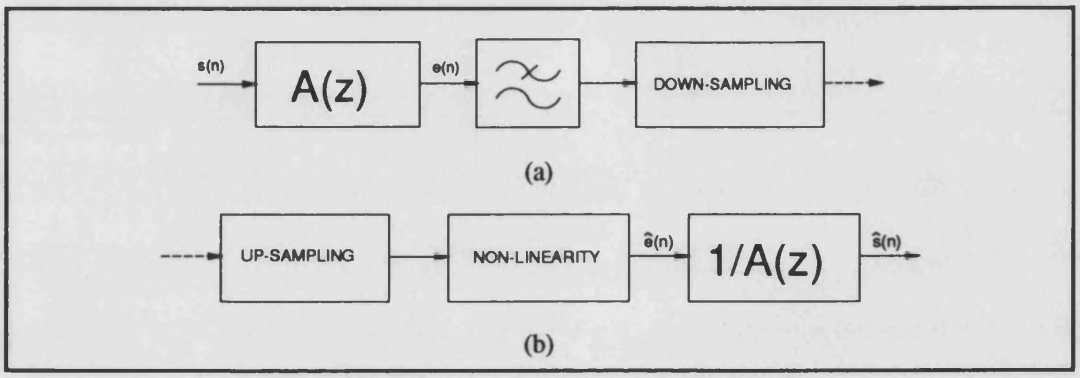

Figure 3.18 *RELP vocoder (a) encoder, (b) decoder.*

At the receiver, the excitation information is decoded and up-sampled, by inserting zeros in between samples, to generate a signal at the original sampling frequency. Higher frequency information is regenerated using a non-linearity, such as a full wave rectifier, before input to the synthesis filter giving synthetic speech.

The computational complexity of this coder was such that it could be easily implemented, in real time, using a single TMS320C20 digital signal processor [9].

This first Baseband RELP vocoder was characterised by a "hoarse" or "breathy" sound and improvements were made by several researchers: Makhoul *et al* [31], Dankberg *et al* [10], Hedelin [20] and Zinser [56]. These mainly concentrated on improving the high

frequency regeneration within the coder. After much research, the sound quality could at best be described as metallic, since no method of high-frequency regeneration was sufficiently accurate.

3.3.3 Residual Pulse Excited (RPE) Vocoder

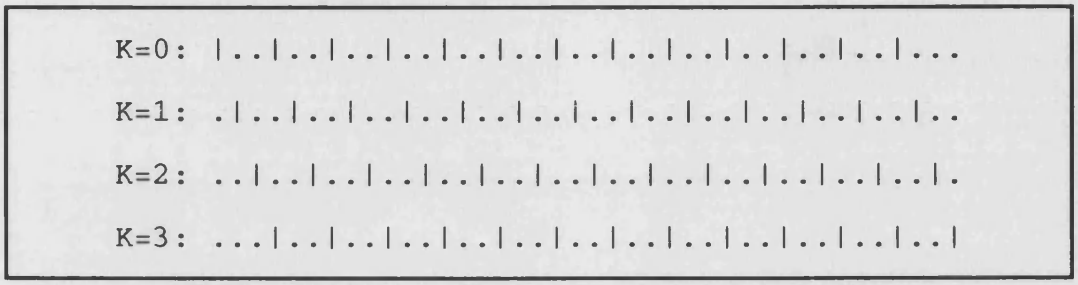

Figure 3.19 *Possible excitation patterns with N=40, Q=13 and M=3.*

The RPE coder [24] is also a member of the class of low complexity coders. The amplitude of one in every M (usually either 3 or 4) residual samples is encoded and transmitted to the decoder. Analysis is performed over adjacent subframes of N samples (normally either 40 or 60) and each subframe is described using Q pulses. Figure 3.19 shows the possible excitation patterns for a subframe containing 40 samples and a spacing of $M = 3$. This gives four possible RPE grid start positions, denoted by K. In this figure, the locations of the pulses are marked by a vertical line, and the zero samples by dots. The optimum grid start position is the one which maintains most residual energy. Each residual frame is transmitted as a frame maximum along with normalised pulse amplitudes.

Performance of the RPE vocoder can be improved with inclusion of a long-term predictor as illustrated in figure 3.20. For satisfactory operation, the encoder analysis LTP must maintain an identical adaptive codebook to the decoder synthesis LTP. This can only be achieved if the encoder contains its own local RPE decoder, since the decoded short and long-term residual is generally significantly different to the actual

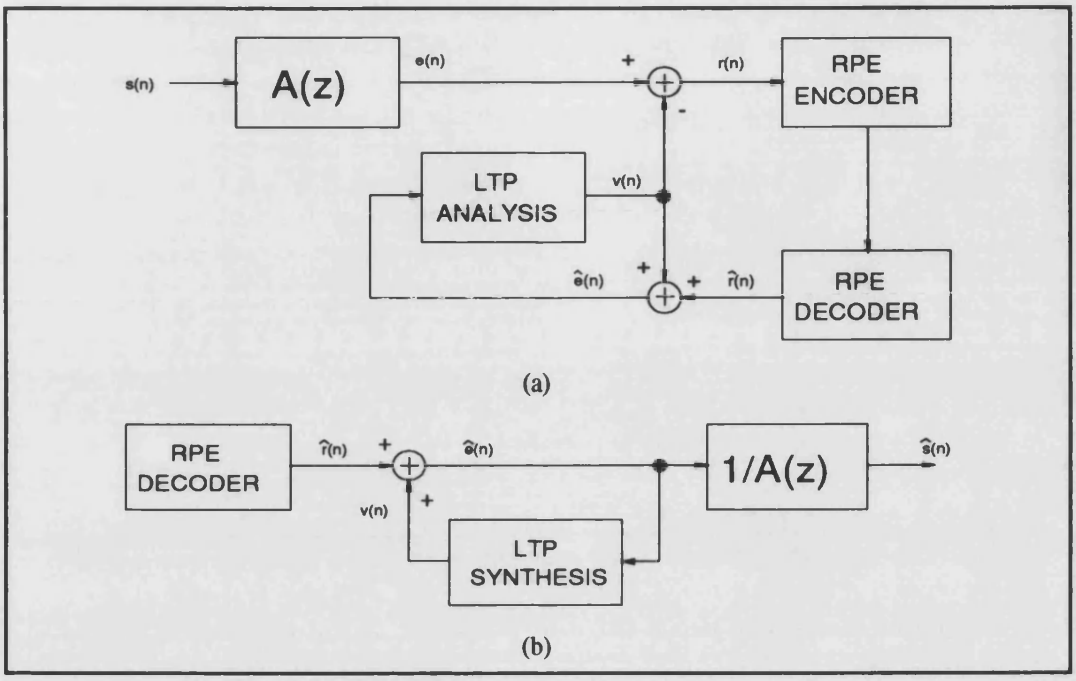

Figure 3.20 *RPE Speech Coder with LTP, (a) Encoder, (b) Decoder.*

signal. Identical codebooks would not be possible if one was based on the actual signal, and the other was based upon the decoded signal. The RPE coder as it is used in the GSM mobile telephone system is described in detail in chapter 4.

3.3.4 Multipulse Excited (MPE) Vocoder

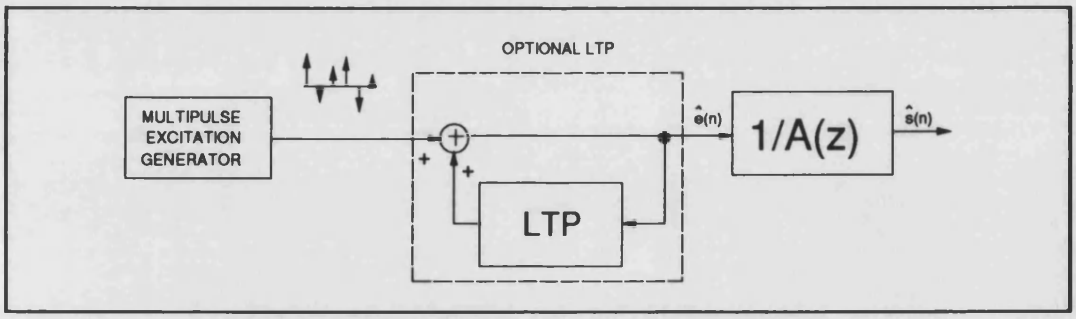

Figure 3.21 *Multipulse Vocoder Incorporating a LTP.*

The Multipulse Excited Vocoder [3], depicted in figure 3.21, is one of the class of high complexity analysis-by-synthesis coders. Its excitation generator describes each subframe excitation as a number of impulses for each frame (usually 8-10). Their

amplitudes and positions are chosen such that the synthetic speech is as perceptually close to the original speech as possible. Thus the transmitted excitation information consists of pulse positions and amplitudes. Multipulse Vocoders often incorporate a LTP, which increases the number of pulses reaching the synthesis filter and hence improves voiced speech quality.

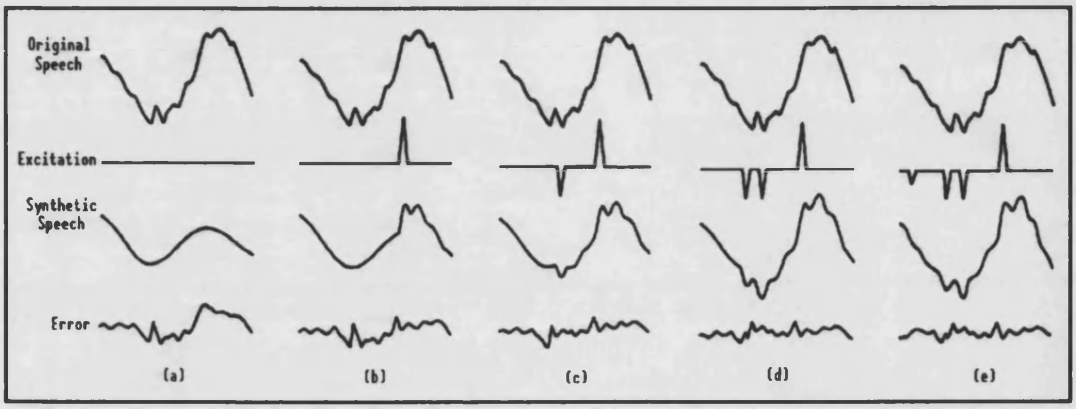

Figure 3.22 *Illustration of the multipulse excitation when between 0 and 4 pulses are used. With 0 pulses the filter has as initial conditions the last outputs of the previous speech frame (after [3]).*

The effect of adding such pulses to the excitation is illustrated in figure 3.22. With no excitation pulses, the speech waveform is given entirely by the memory of the synthesis filter from the previous subframe. As more excitation pulses are added, the synthetic speech resembles the input speech far more closely, at the expense of a higher data rate.

3.3.5 Code Excited Linear Prediction (CELP) Vocoder

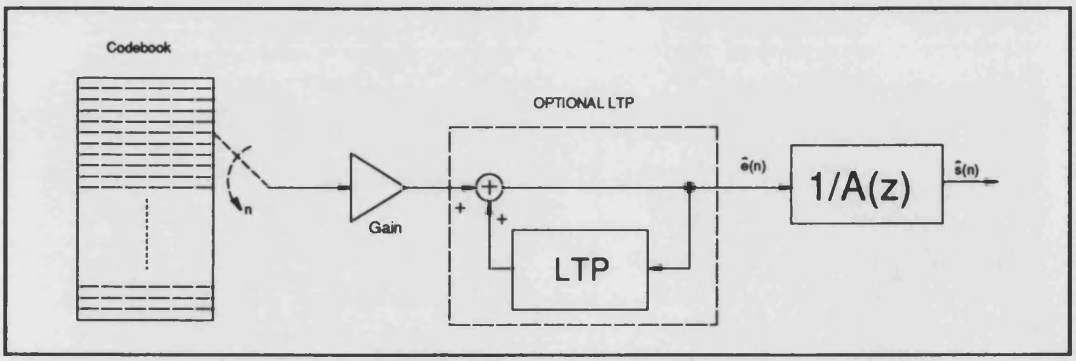

Figure 3.23 *The CELP Vocoder.*

In the code excited linear prediction (CELP) vocoder, depicted in figure 3.23, the excitation generator consists of a codebook of stored sequences. The optimum excitation sequence is determined by an exhaustive search of all possible codebook entries. The codebook entry index n and the associated gain giving the least error between original and synthetic speech are transmitted to the decoder. Again the practical CELP coder is normally combined with a LTP.

The original CELP coder [48] used a codebook of 1024 ensemble entries each 40 samples long of gaussian white noise of unit variance. An alternative way to implement a codebook, is to have a long sequence, where each candidate entry is an overlapping subsequence. Codebooks of this nature in which neighbouring candidates overlap for all but two samples, have performance as high as for codebooks containing fully independent ensembles [23]. This has advantages both from the point of view of storage and of computational complexity. The CELP coder has gained widespread acceptance and a version has been adopted as the U.S. Federal standard for low bit rate speech coding at 4800bit/s [6].

Other CELP codebooks worthy of mention are: (a) Sparse, where the codebook consists of gaussian random ensembles in which most of the excitation sample values have been set to zero. The number of non zero excitation samples in each ensemble is referred to as the weight. Typical weight values are in the range of 3 to 6 for excitation vectors of length 40. Each ensemble in the codebook has the same weight. (b) Ternary, where a ternary excitation vector is derived from a sparse excitation vector by taking only the signs of the non zero excitation amplitudes. Each non zero sample retains its sign and has unity magnitude.

CELP vocoders developed within this thesis all use sequential codebooks with overlapping candidate excitation vectors. The search through the codebook for the optimum excitation vector and calculation of the corresponding gain is very similar to

that of the conventional LTP. The starting point is target vector $\tilde{s}(n)$, given by equation 3.60. Alternatively if the coder incorporates a LTP, the starting point is target vector $\tilde{s}'(n)$, which is the difference between the original target vector and the contribution from the LTP stage

$$\tilde{s}'(n) = \tilde{s}(n) - \gamma v_l(n - d_l) * h(n) \qquad 3.66$$

The following theory is applicable to both target vectors $\tilde{s}(n)$ and $\tilde{s}'(n)$. The mean squared error between target vector and synthetic speech contribution from codebook stage is given by

$$E(d_F, \phi) = \sum_{n=0}^{N-1} \tilde{s}^2(n) - \phi \sum_{n=0}^{N-1} \tilde{s}(n)\,[v_F(n - d_F) * h(n)] \qquad 3.67$$

where ϕ is the fixed codebook gain, v_F is the fixed codebook buffer and d_F is the fixed codebook index. For a given codebook index d_F, minimisation of E with respect to ϕ gives

$$\phi = \frac{\sum_{n=0}^{N-1} \tilde{s}(n)\,[v_F(n - d_F) * h(n)]}{\sum_{n=0}^{N-1} [v_F(n - d_F) * h(n)]^2} \qquad 3.68$$

and

$$E(d_F) = \sum_{n=0}^{N-1} \tilde{s}^2(n) - E'(d_F) \qquad 3.69$$

where

$$E'(d_F) = \frac{\left(\sum_{n=0}^{N-1} \tilde{s}(n)\,[v_F(n - d_F) * h(n)]\right)^2}{\sum_{n=0}^{N-1} [v_F(n - d_F) * h(n)]^2} \qquad 3.70$$

The optimum codebook index d_F is that which maximises $E'(d_F)$ in equation 3.70. The predictor gain ϕ is then calculated from equation 3.68.

3.3.6 The Self-Excited Vocoder (SEV)

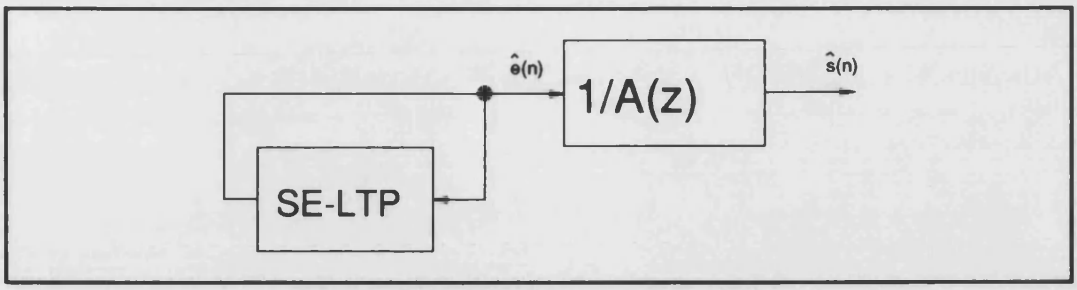

Figure 3.24 *The Self-Excited Vocoder (SEV).*

The Self-Excited Vocoder (SEV), depicted in figure 3.24, was introduced by Rose and Barnwell in 1986 [42]. Synthesis filter excitation is derived from the past history of the excitation signal itself. This is achieved using a conventional LTP without any input! Thus the output from the self-exciting LTP (SE-LTP) is based upon its own *previous* output. Thus in operation, the SEV applies no bits to coding the residual at all. To commence operation a SE-LTP history must be established. Early methods used the actual linear predictive analysis residual for the first few frames before switching to SEV operation. This served to demonstrated the technique, but is not suitable for a practical mobile radio scheme. Later studies spoke of filling the history buffer with a zero mean, unit variance gaussian distributed sequence at the start of the coding session. This provided a pseudo history and operation could commence.

Optimisation of SE-LTP parameters is very similar to the optimisation of conventional LTP parameters. The starting point is target vector $\tilde{s}(n)$, given by equation 3.60. The mean squared error between target vector and synthetic speech contribution from the SE-LTP stage is given by

$$E(d_s,\zeta)=\sum_{n=0}^{N-1}\tilde{s}^2(n)-\zeta\sum_{n=0}^{N-1}\tilde{s}(n)[v_s(n-d_s)*h(n)] \qquad 3.71$$

where ζ is the SE-LTP gain, v_s is the SE-LTP adaptive codebook buffer and d_s is the SE-LTP delay. For a given delay d_s, minimisation of E with respect to ζ gives

$$\zeta = \frac{\sum_{n=0}^{N-1} \tilde{s}(n)\,[v_s(n-d_s)*h(n)]}{\sum_{n=0}^{N-1} [v_s(n-d_s)*h(n)]^2} \qquad 3.72$$

and

$$E(d_s) = \sum_{n=0}^{N-1} \tilde{s}^2(n) - E'(d_s) \qquad 3.73$$

where

$$E'(d_s) = \frac{\left(\sum_{n=0}^{N-1} \tilde{s}(n)\,[v_s(n-d_s)*h(n)]\right)^2}{\sum_{n=0}^{N-1} [v_s(n-d_s)*h(n)]^2} \qquad 3.74$$

The optimum SE-LTP delay d_s is that which maximises $E'(d_s)$ in equation 3.74. The SE-LTP gain ζ is then calculated from equation 3.72.

The Self-Excited Vocoder is the subject of this thesis, with the aim of assessing its suitability as the basis of a mobile telecommunications speech coder for operation below 8kbit/s. Chapter 5 introduces the SEV with a simple low complexity version. Chapter 6 introduces the higher complexity analysis-by-synthesis SEV and chapter 7 studies combinations of LTPs and SE-LTPs to form practical speech coders.

Chapter 4

GSM Speech Coding

Chapter 4

GSM Speech Coding

In October 1985, under the direction of GSM, a Speech Coding Experts Group (SCEG) was formed with participation from Britain, Finland, Federal Republic of Germany, France, Italy, The Netherlands, Norway, and Sweden. Their task was, firstly to specify the design requirements of a suitable codec, secondly to assess the suitability of all candidate codecs and finally to select one basic scheme for further optimisation to the stage where a detailed specification of the algorithm could be produced.

Section 4.1 describes the speech coder requirements, section 4.2, the speech coder trials, with sections 4.3, 4.4 and 4.5 describing the encoder algorithm, the decoder algorithm and the air interface respectively. The complete speech coder has been implemented in non-real time. Much effort was spent assessing the performance of this speech coder, this is covered in the final section 4.6, which is subdivided into sections covering overall operation, noisy channel performance, including a study of the benefits of the LTP stage, tandem performance and a study of the benefits of the coder preemphasis and deemphasis functions.

4.1 Speech Coder Requirements

Speech Quality: from the subscribers point of view, any speech encoder should give at least as good speech quality as the first generation 900MHz analogue mobile telephones. In particular, the coding algorithm should be robust to variations in voice spectra and levels. Differences in spectra are due to different talkers, microphones and transmission effects of the various telephone networks. The speech encoder should be robust to environmental noise and multiple speakers and capable of operation in noisy environments such as lorry cabs and fast cars. The final requirement for speech quality is that conversation should be intelligible when mobile to mobile connection is made and two coder/decoder systems are cascaded together.

Gross Bit Rate of 16kbit/s: This was chosen as a compromise between the expected performance of available coders and the required spectral efficiency. In anticipation of future mobile telephone demand and future advances in real-time speech encoding, provision was to be made to accommodate a "half-rate" codec, when it became available, which would double the spectral efficiency without any modification to the radio sub-system. The sampling rate would remain at 8kHz to allow simple interfacing to the PSTN.

Transcoding: The speech codec would have to operate satisfactorily with speech already encoded as A- or μ- law PCM.

Transmission of Non-Voice Signals: Coding schemes operating with bit rates of the order of 16kbit/s exploit properties of the speech signal to give good perceptual quality. It was decided that a compromise scheme to accommodate both speech and voiceband data would most likely reduce the speech quality. Thus the coder would be optimised for transmission of speech and transmission of voiceband data would use special terminal adaptors. However, the speech coder should be capable of acceptable transmitting the audio tones (such as dialling, ringing and busy etc) to the subscriber by the network.

Delay: In any telephone connection, reflections occur in the impedance mismatches at the four to two wire conversions at the extremities. Excessive delay in both the speech coder and the radio sub-system will result in echos causing serious disturbance to customers. It was decided to set an upper limit of 65ms delay for the coder/decoder combination along with the use of echo control devices.

Computational Complexity: This requires that the algorithm could be implemented on a single VLSI chip having minimum power consumption.

4.2 Speech Coder Trials

Initially over 20 codecs with application to digital mobile radio were reported in the participating countries. After initial selection on an individual country basis, 6 algorithms were presented to the pan-European Speech Coding Experts Group. Subjective testing of all 6 candidate coders started in September 1986 [8].

In order to assess practical factors such as transmission delay and computational complexity, all entries had to be presented as real-time hardware laboratory models. This ensured a certain maturity in algorithm development. Each codec was to have an equal gross bit-rate of 16kbit/s. Within this total, bits could be allocated to speech or channel coding as designers wished, to give satisfactory performance with bit error rates up to 1%.

The presented coders could be grouped into two classes: Four of them were Sub-Band coders and the other two were Linear Predictive coders. The four Sub-Band coding techniques are described in [28],[41],[32] and [18]. The Linear Predictive coders were a Simplified Regular Pulse Excited LPC Codec (RPE-LPC) [52], developed by Philips Kommunikations Industrie AG, Nurnberg, Fed. Rep. Germany, and a Multipulse Excited Codec with Long Term Prediction (MPE-LTP) [15], developed by IBM Laboratory, La Gaude, France.

The subjective testing was performed in seven different laboratories by subjects listening to their own native language. In order to ensure identical test conditions, the source tapes from Britain, Italy, France, Federal Republic of Germany, Norway, the Netherlands and Sweden, were all processed on the hardware algorithm models at a joint session hosted by CSELT. in Italy. Recordings of coded/decoded speech were then returned to their respective laboratories for evaluation.

The conclusions of this evaluation [35] were:

1. The Sub-Band coders were outclassed by the Linear Predictive coders in terms of speech quality.
2. The RPE-LPC coder had the best average quality and was a member of the group of coders of lowest computational complexity.
3. The MPE-LTP coder was a close second to the RPE-LPC coder in terms of speech quality but had 3 times the computational complexity.
4. The RPE-LPC was chosen as the starting point for further studies to determine whether or not it would be advantageous to include the Long Term Predictor of the MPE-LTP design

These further studies concluded that the net bit rate of the codec could be reduced from 14.77kbit/s to 13.0 kbit/s by adding a long term predictor [51] while still maintaining equivalent sound quality. This freed more bits for channel error protection. A detailed specification down to precise computational details was produced [17] and this has been implemented in non real time by the author.

4.3 The Encoder Algorithm

The RPE speech coder has already been briefly described in chapter 3. The encoder algorithm divides naturally into 5 sections: Pre-Processing, LPC Analysis, Short Term Analysis Filtering, Long-Term Prediction and RPE Encoding. A detailed block diagram of the encoder is given in figure 4.1. This block diagram is an extension to that of figure 3.20.

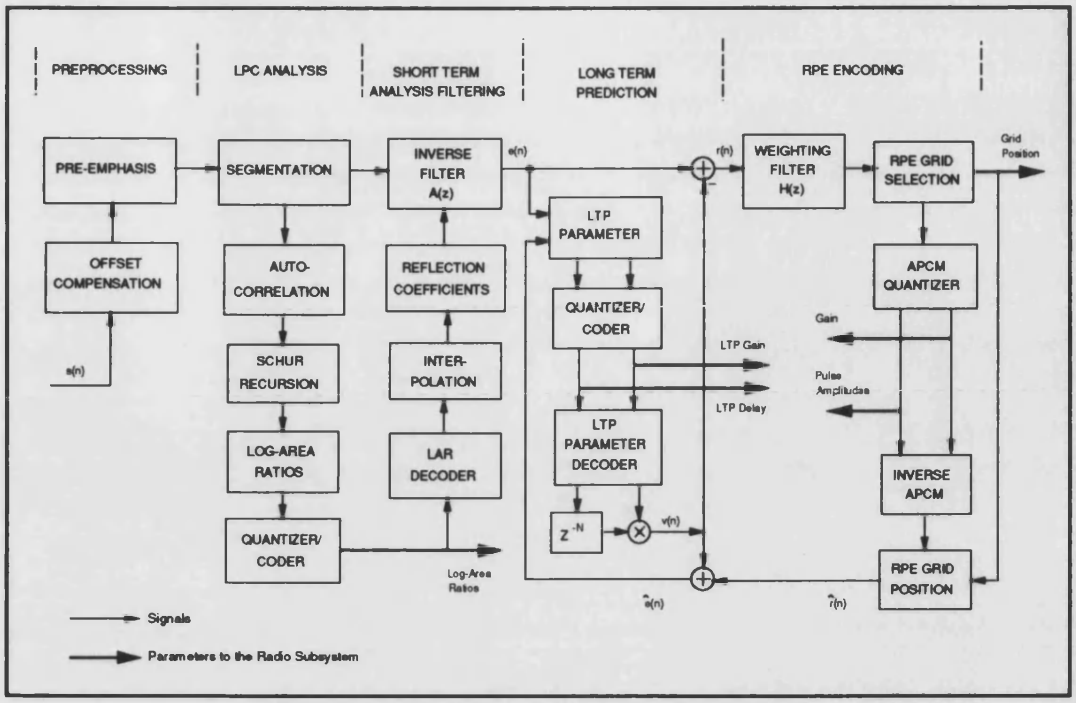

Figure 4.1 *Block Diagram of the RPE-LTP Encoder*

4.3.1 Preprocessing

Offset compensation prevents any D.C. component being translated into an annoying side tone by the decoder high frequency regeneration process. This is achieved by a single tap high pass filter which uses 32 bit arithmetic. The signal is then preemphasised by a first order FIR filter enabling more efficient fixed point processing in the subsequent stages.

4.3.2 LPC Analysis

In the segmentation buffer, the speech signal is divided into non-overlapping segments of length 20ms (160 samples). A new, rectangular windowed, autocorrelation LPC analysis is performed for each frame and 8 reflection coefficients are calculated by the processes of auto-correlation and the LeRoux-Guegen method [30]. The first version RPE codec (before the inclusion of the long-term predictor) used Hamming windowed, overlapping segments for the LPC analysis, this approach was abandoned since it gave no improvement in speech quality, along with an increase in overall delay.

The reflection coefficients which have range $-1 \leq r_i \leq +1$ and are converted into Log-Area-Ratios due to their favourable quantisation characteristics [53]. This conversion is strictly defined as

$$LAR_i = \log_{10}\left(\frac{1+r_i}{1-r_i}\right) \qquad 4.1$$

Since it is the companding characteristic that is important, the following segmented approximation is used

$$LAR_i = \begin{cases} r_i & |r_i| < 0.675 \\ sign[r_i] \times [2|r_i| - 0.675] & 0.675 \leq |r_i| < 0.950 \\ sign[r_i] \times [8|r_i| - 0.950] & 0.950 \leq |r_i| \leq 1.000 \end{cases} \qquad 4.2$$

This reduces highly computationally complex division and logarithm operations to much more simple add, multiply and compare. The inverse of this transformation is given by

$$\hat{r}_i = \begin{cases} \hat{LAR}_i & |\hat{LAR}_i| < 0.675 \\ sign[\hat{LAR}_i] \times [0.500|\hat{LAR}_i| + 0.337500] & 0.675 \leq |\hat{LAR}_i| < 1.225 \\ sign[\hat{LAR}_i] \times [0.125|\hat{LAR}_i| + 0.796875] & 1.225 \leq |\hat{LAR}_i| \leq 1.625 \end{cases} \qquad 4.3$$

Due to the different dynamic ranges and different asymmetric amplitude distributions [51], the Log-Area-Ratios are quantised with the bit allocation shown in table 4.1.

LAR No. i	Bits/LAR
1,2	6
3,4	5
5,6	4
7,8	3

Table 4.1 *Bit Assignment of the LAR Coefficients.*

4.3.3 Short-Term Analysis Filtering

Within the encoder, the 8 Log-Area-Ratio values are converted to the 8 reflection coefficients of the short-term analysis filter using identical code as in the decoder. Firstly, the current and previous frames Log-Area-Ratios are interpolated linearly within a transition period of 5 ms to avoid spurious transients. Secondly, they are reconverted into reflection coefficients r_i' of the lattice type analysis filter. The 160 preemphasised samples of current frame input speech are then filtered producing 160 samples of residual signal $e(n)$.

4.3.4 Long-Term Predictor

The LTP loop is used to compute the estimate $v(n)$ of the residual signal $e(n)$ from the history of the reconstructed excitation signal $\hat{e}(n)$. The speech frame is divided into 4 sub-frames of 40 samples for input to the LTP filter, which is characterised by the gain γ and the delay d according

$$v(n) = \gamma' \hat{e}(n-d) \tag{4.4}$$

where γ' denotes the quantised version of γ. Along with parameter d this is calculated every 40 samples (5ms). Each 160 sample segment of the residual $e(n)$, starting with e_{n_0}, has sub-segments $d(n_0 + j \cdot 40 + i)$ $(j = 0,1,2,3; \quad i = 0...39)$. Then the error function $E(d)$ is calculated according to

$$E(d) = \sum_{i=0}^{39} e(n_j + i)\hat{e}(n_j + i - d); \qquad \begin{array}{l} n_j = n_0 + 40j \\ j = 0,1,2,3 \\ d = 40..120 \end{array} \tag{4.5}$$

The optimum delay value d_{opt} is that for which this function is maximum

$$E(d_{opt}) = \max\{E(d); \quad d = 40..120\} \tag{4.6}$$

The lower limit for the delay is $d = 40$, thus the optimum delay d does not necessarily correspond to one pitch period of the speech signal, but could be a multiple. The long-term predictor gain γ for the j-th sub-segment is calculated by

$$\gamma = \frac{E(d)}{\sum_{i=0}^{39} \hat{e}^2(n_j + i - d)} \qquad 4.7$$

The LTP parameters γ and d are encoded with 2 and 7 bits, respectively. It should be noted that the LTP for this speech coder is very simple, having only integer sample delay values and the delay range not extending below 5ms.

4.3.5 RPE Encoding

An 11 tap FIR "weighting filter" is applied to each sub-segment of 40 samples of the long-term filtered residual $r(n)$. Conventional convolution of a sequence having 40 samples with an 11-tap impulse response would produce 50 samples. To overcome this, the weighting filter algorithm produces the 40 central samples of the conventional convolution operation. For notational convenience, denote the block filtered subsegment by $x(n), n = 0..39$. For the next step the filtered signal $x(n)$ is down-sampled by a ratio of three giving three interleaved sequences of lengths 14, 13 and 13 samples. The sequence of length 14 separates into two sequences of 13 samples, one containing the first 13 and the other containing the last 13 samples.

$$x_m(i) = x(n_j + m + 3i) \qquad \begin{matrix} m = 0,1,2,3 \\ i = 0,1..12 \\ n_j = n_0 + 39j \end{matrix} \qquad 4.8$$

With n_j defining the beginning of the j-th sub-segment and m denoting the phase of the decimation grid. The optimum sequence $x_M(i)$ is the one having the most energy as given by

$$E(m) = \max \sum_{i=0}^{12} x_m^2(i) \qquad m = 0,1,2,3. \qquad 4.9$$

Finally the selected RPE-sequence is quantised by block adaptive PCM (APCM). Each block of 13 samples is normalised by its block maximum x_{max}. The samples are then

quantised uniformly with 3 bits, the block maximum is coded logarithmically with 6 bits, and the grid position M is coded with 2 bits. The overall bit allocation of the GSM. RPE-LTP speech coder is given in table 4.2.

Parameter	Number of Bits
8 LAR Coefficients $LAR(i)$	36
4 LTP Gains γ	8
4 LTP Delays d	28
4 RPE Grids M	8
4 Block Maxima x_{max}	24
52 RPE Samples $x_M(i)$	156
Total Bits per 20ms Frame	260

Table 4.2 *Bit Allocation (Bit Rate = 13.0 kbit/s)*

4.4 The Decoder Algorithm

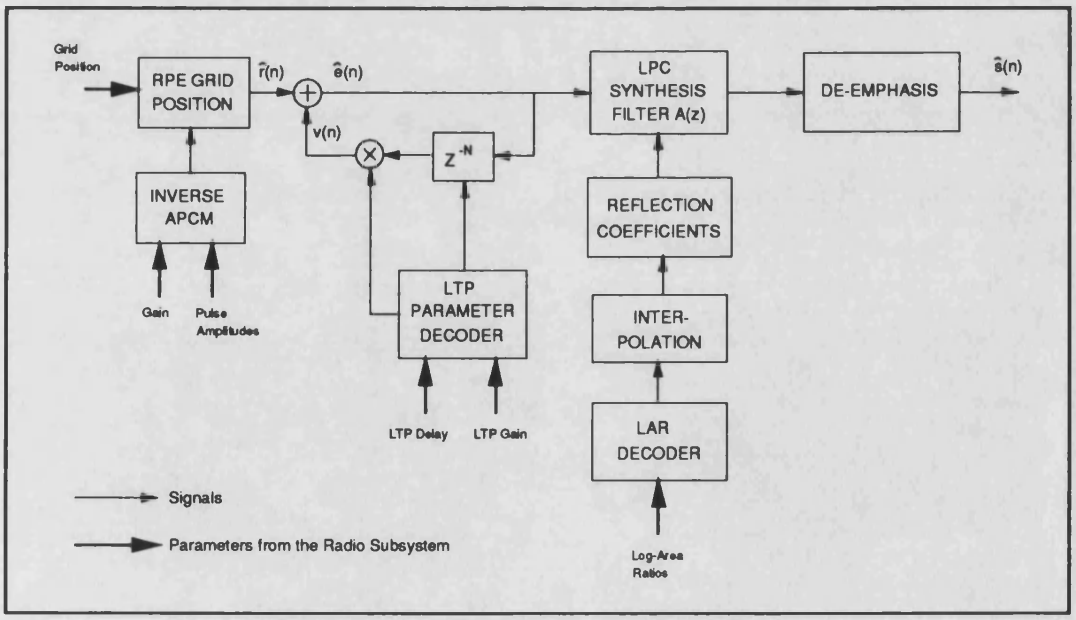

Figure 4.2 *Block Diagram of the RPE-LTP Decoder.*

The decoder is very simple in comparison with the encoder, its block diagram is shown in figure 4.2. The received RPE parameters M, $x_M(i)$ and x_{max} are decoded and used to reconstruct the excitation $\hat{r}(n)$ of the long-term synthesis filter. The sample rate of the RPE samples $x_M(i)$ is increased by a factor of 3 by inserting zero samples and by placing the non-zero samples in the correct grid position M. The Long-term synthesis filter re-creates the excitation $\hat{e}(n)$ of the short-term synthesis filter, again a lattice type. After synthesis filtering, the output speech is de-emphasised by a filter inverse to the original pre-emphasis filter.

4.5 The GSM Air Interface

This chapter would not be complete without a description of the other components of the air interface. The following sections briefly describe the channel coder, the transmission data structure, the modulation scheme and multipath equalisation.

4.5.1 Channel Coder

The original design requirement for the speech coder trials was that it should work acceptably, with bit error rates up to 1%, with an overall bit rate of 16kbit/s. In practice, greater error rates are likely to be encountered and the channel coder adds redundancy, increasing the overall data rate to 23kbit/s improving the overall error performance.

The channel coder operates in 4 stages:-

1. The output from the speech coder consists of 260 information bits per frame, these have been ordered in terms of there subjective importance. These are divided into 182 class 1 error protected bits, and 78 class 2 unprotected bits. The fifty most important bits are protected by 3 parity bits, which will be used for error detection in the decoder. If an error occurs in these bits, the entire frame will be considered bad, and the previous frame will be repeated. This can happen up to 4 times, with progressive attenuation each frame.

2. The class 1 information and parity bits are re-ordered and tail bits are added, this gives 189 bits describing the original class 1 information bits.

3. These 189 bits are then encoded with an error correcting half rate convolutional code giving 378 bits. The 78 unprotected class 2 bits are then added giving a total of 456 bits describing the complete speech frame.

4. These 456 bits are re-ordered and re-distributed over 8 time-slots in consecutive TDMA frames. Even numbered bits are used in the first 4 time-slots and odd numbered bits in the last 4 time-slots. Thus any time slot carries 57 bits of data from one speech frame and 57 bits of data from the next speech frame, where the bits from the most recent speech frame are always the even numbered bits.

4.5.2 Transmission Data Structure

Channel Spacing	200kHz
Modulation	GMSK
Data Transmission Rate	270.833kbit/s
Number of Channels/Band	8 (16)
User Data Rate (Nominal)	16 (8)kbit/s
TDMA Frame Period	4.62ms
Time Slot Duration	0.58ms

Table 4.3 *Basic Air-Interface Parameters*

The specific parameters selected for the GSM air interface are shown in Table 4.3. The basic traffic rate allows 8 channels to be accommodated on a single RF carrier. With an eye to the future, the specification allows two channels to be interleaved on the same frame. This facility will double the traffic capacity when an 8kbit/s toll quality voice coder is available. The planned date for this introduction is approximately 1995. Figure 4.3 shows the basic frame structure and the time-slot organisation for a traffic or signalling channel.

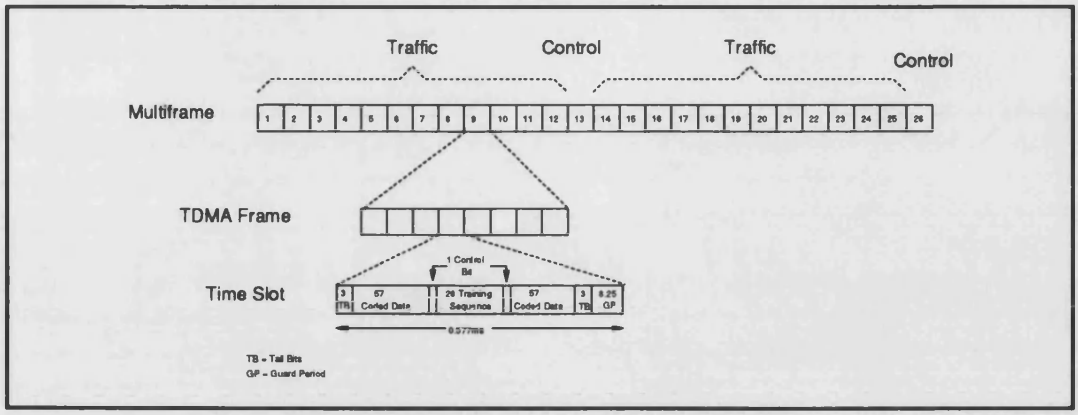

Figure 4.3 *Channel and Time Slot Organisation.*

Each time-slot lasts 0.577ms and comprises 148 bits with an 8.25 bit guard period between the slots. The traffic carried by the slot is divided into two 57 bit blocks, each containing data from separate speech frames. Thus 8 such slots are needed to convey the 20ms of speech data but each slot is actually carrying two sets of data simultaneously. Thus 4 consecutive time slots provide 456 traffic bits in 18.5ms accommodating 20ms of speech. The additional 1.5ms, when summed over 24 frames, provides the two control frames in the multiframe.

4.5.3 Modulation

The modulation method of the GSM system had to meet 4 requirements:

1. Frequency translation into the correct band,
2. Relatively narrow bandwidth to allow good spectral efficiency,
3. Constant envelope to allow the use of simple and efficient power amplifiers, Eg. class C,
4. Low out-of-band radiation such that adjacent channel interference is low.

Minimum shift keying (MSK), which is a binary digital frequency modulation with index 0.5, achieves the first three, unfortunately it fails on the fourth requirement having excessive out-of-band radiation.

Pre-filtering of the modulating signal reduces the out-of-band radiation whilst retaining the constant envelope property. A gaussian pre-modulation filter gives Gaussian MSK (GMSK), which has all the required properties. A bandwidth data-rate product of 0.3 was selected [38].

4.5.4 Multipath and Equalisation

At the GSM frequency band, radio waves refract poorly which could lead to many shadow areas when either mobile or base station transmits. This is compensated for by reflections from buildings, hills, high-sided vehicles etc, which help to fill in the shadows. Many different reflections can reach the same point and even when there is a direct path it is not unusual for strong reflections to be received as well. The radio paths taken by the reflections are longer than the direct path and the difference in path length can be several bit periods.

Unless some corrective action is taken, GSM would be unusable in all but very small (micro) cells. In the centre of each time slot is a training sequence of 26 bits, which is used by the receiver to calculate the multipath delay profile of each burst. This information is used by a Viterbi equaliser to compensate for the multipath propagation. These conditions generally change considerably from one frame to the next, particularly with high speed mobiles. Thus, the training sequence is placed in the centre of each burst, minimising the change in multipath delay profile from its measurement to either extreme of the burst.

4.6 Results

The complete GSM pan-European speech coder was implemented (in non real time) from its specification [17] using the "C" language. The nature of the implementation is described in appendix 1.2, where the encoder/decoder is implemented as one program along with a separate decoder only program. The parameter quantisation used was exactly as specified by GSM. In default mode, the implementation worked exactly as the specification described, however command line switches could be set disabling the

long-term predictor functions, disabling the pre-emphasis and deemphasis functions, or to output "trace waveforms" showing the passage of a signal through the algorithm. The objective results quoted in this results section were measured over 53 seconds of speech consisting of 20 sentences from the Harvard list of phonetically balanced sentences [21] spoken by 10 males and 10 females.

4.6.1 Overall Operation

The overall operation of the complete speech coder is demonstrated figuratively by showing the passage of a short signal from encoder speech input to decoder speech output. These results are shown in figures 4.4 and 4.5. The test signal used in these plotted waveforms was the utterance "seven" by a male speaker (the author). As shown in the top trace of figure 4.4, the word divides into three distinct segments of widely differing characteristics. The word begins unvoiced (the *s*), with a low level, random noiselike appearance. It then voice fricates, a combination of voicing and unvoicing (the *v*), and the waveform shows a combination of some periodicity and some randomness. It finishes entirely voiced (the *en*), and the waveform shows an obvious periodicity.

Figure 4.4 shows the detailed operation of the encoder. The top trace, is the input speech as has already been described. The second trace, shows the input speech after short-term analysis filtering. It should be noted that this trace has been amplified by 13dB compared to the top trace, showing the residual has small amplitude compared to that of the input speech. The differences between the residuals of the three segments are very noticeable. The unvoiced sound has been attenuated to about half its original amplitude whilst retaining its appearance. The middle segment, which was the voiced frication, now appears as a noiselike waveform with superimposed pitch pulses. The final segment has almost disappeared leaving only a few small pitch pulses. Significant redundancy has been filtered out of the second and third segments by the analysis filter.

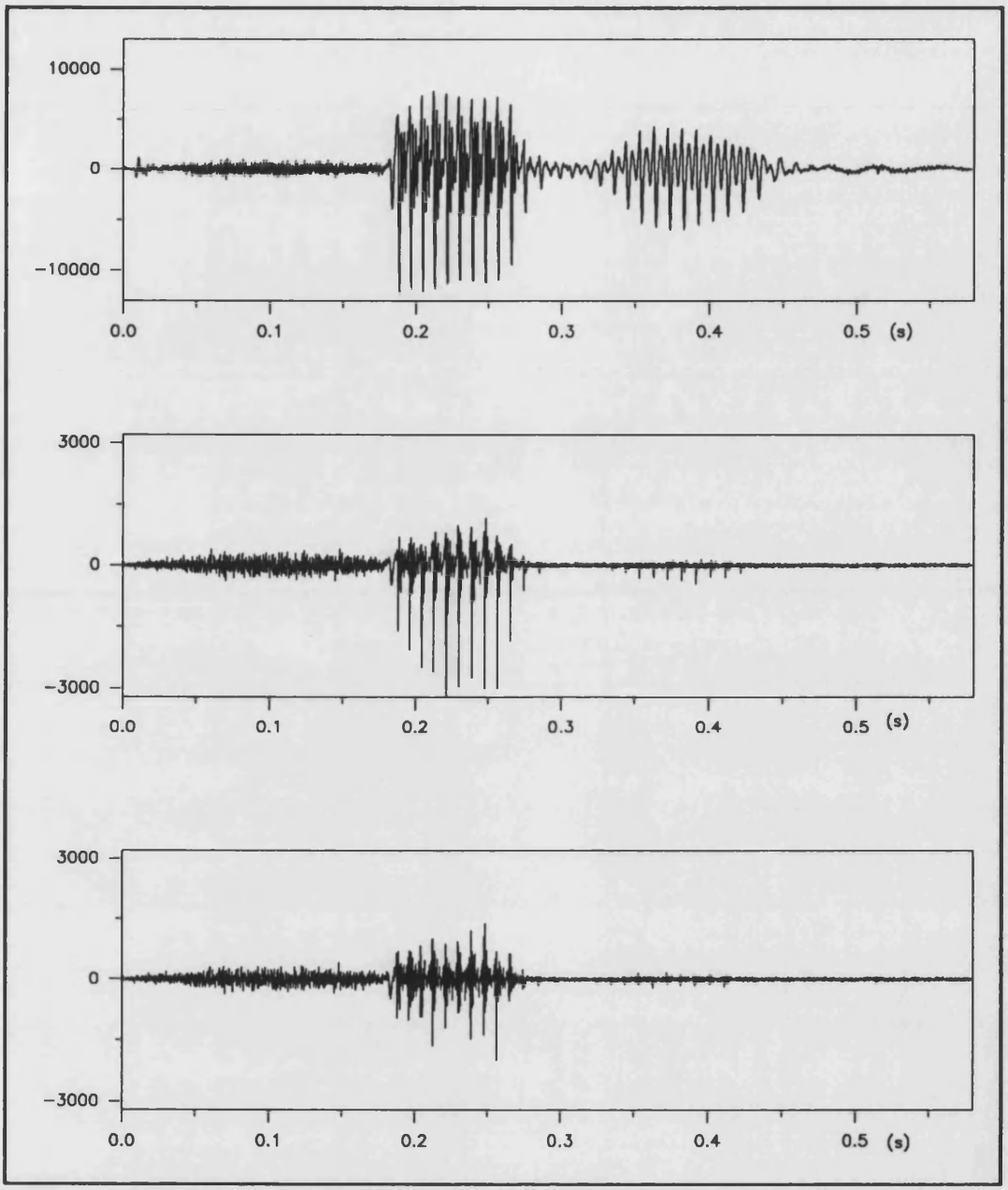

Figure 4.4 *GSM Pan-European speech encoder waveforms, top: input speech ("Seven"), middle: waveform after short-term analysis filtering, bottom: waveform after long-term analysis filtering.*

The bottom trace of figure 4.4, shows this residual after long-term analysis (or pitch) filtering. The first segment contains no periodicity and is unaffected by this filtering operation. The fricated middle segment has a significant reduction in the amplitude of its pitch pulses. The third voiced segment has almost entirely lost all traces of previous

pitch pulses and now has the appearance of very low-level noise. The long-term analysis filter has successfully removed much of the structure from the short-term analysis filter residual. This signal is now encoded as regular pulses for transmission to the decoder.

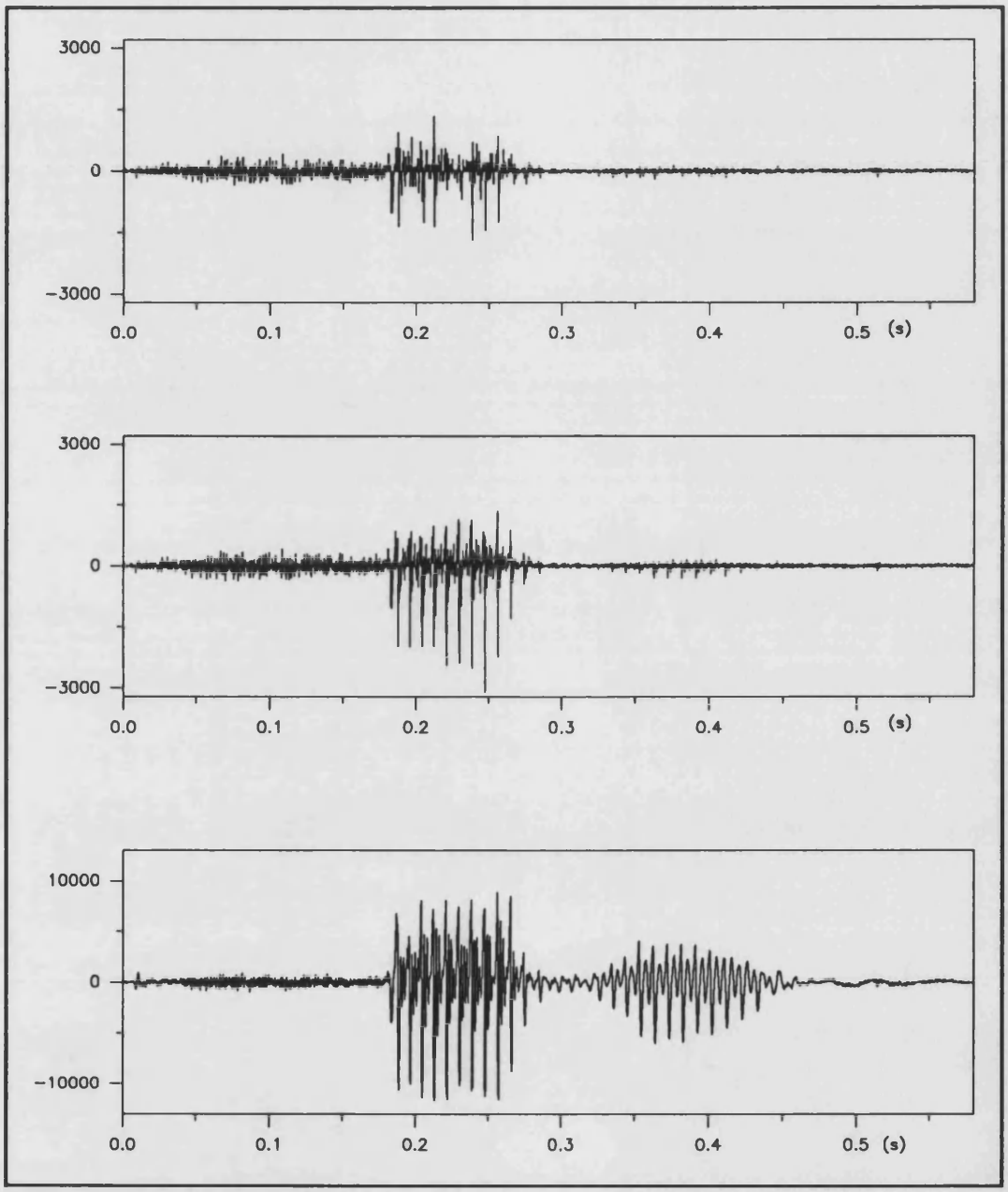

Figure 4.5 *GSM Pan-European speech decoder waveforms, top: reconstructed transmitted residual, middle: waveform after long-term synthesis filtering, bottom: output speech waveform after short-term synthesis filtering.*

Figure 4.5 shows the detailed operation of the decoder. The top trace shows the reconstructed transmitted residual. This is the regular pulse excitation representation of the bottom trace of figure 4.4, which is the short and long- term analysis filtered input speech from the encoder. The overall envelope of the signal has been lost and it appears to be made up entirely of impulses rather than the noiselike waveform of its encoder counterpart. This is because only one out of three sample residual pulse are kept by the residual pulse encoding.

The middle trace is the signal after long-term synthesis filtering. This process recreates much of the residual structure, filling in many of the zeros between the impulses of the previous waveform. This trace shows much similarity with its encoder counterpart. The bottom trace shows the output speech waveform after short-term synthesis filtering. This is a very near toll quality reproduction of the original speech.

To assess the benefit of the LTP, it was disabled from the speech coder. Overall operation on the same short utterance "Seven" is shown in figure 4.6. The middle trace shows reconstructed speech from the coder without the long-term predictor. The general form of the input waveform is maintained but the coder fails to exactly recreate the waveforms periodicity. This is most obvious with the amplitude variation of the peaks in the middle fricated segment. This coder distorted noticeably with periodic sounds such as the tw*elve*, and l*athe* etc. The bottom trace shows the normal output with LTP, this is a notable improvement. The periodicity of the output waveform is much more faithfully reproduced. The third voiced segment shows negligible distortion whereas the middle frication shows only very slight distortion in the amplitudes of its peaks.

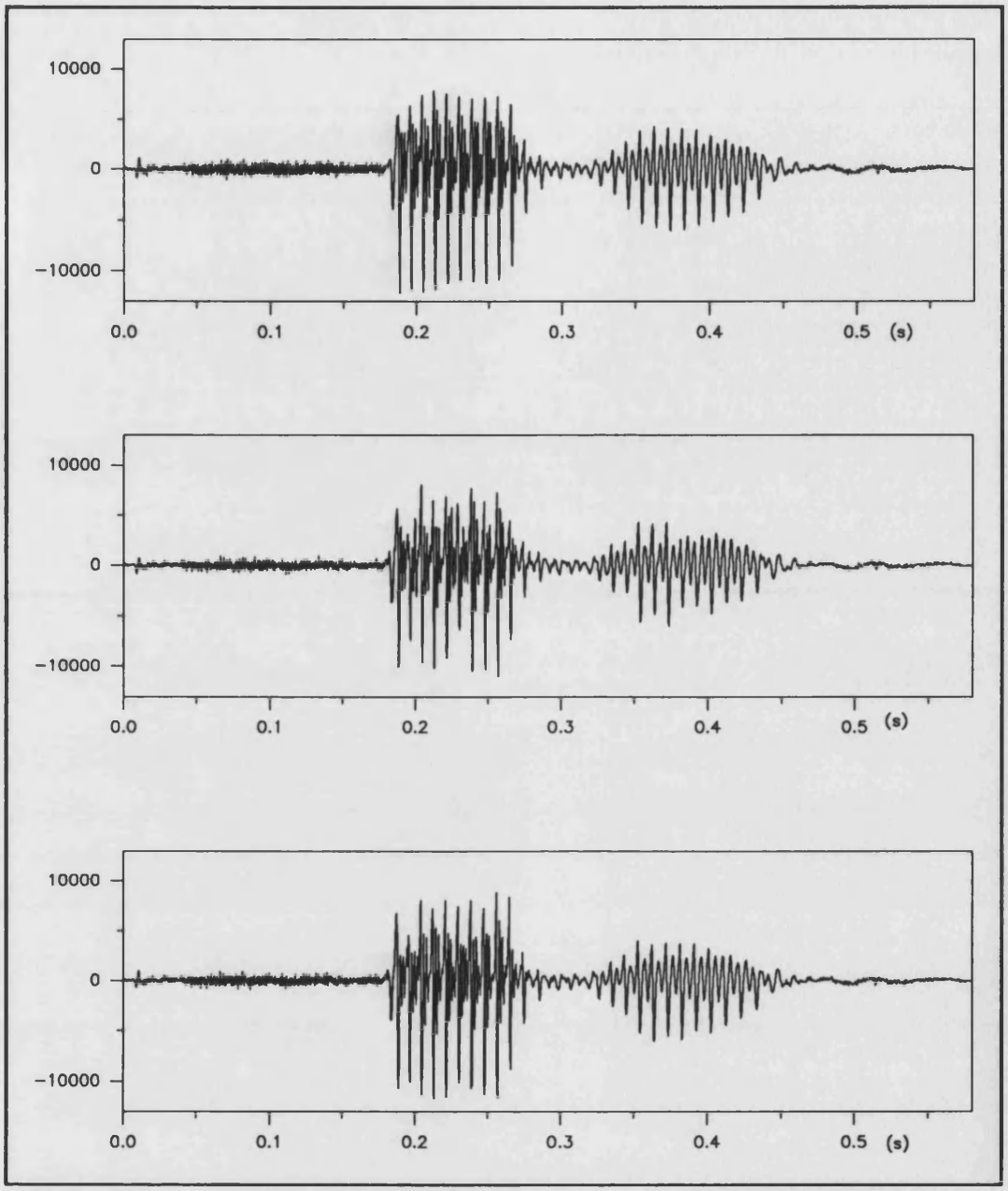

Figure 4.6 *Overall operation of GSM Pan-European speech coder, illustrating the benefit of the long-term predictor, top: input speech ("Seven"), middle: output speech waveform without LTP, bottom: output speech waveform with LTP.*

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)
With LTP	9.44	2.55
Without LTP	7.41	2.60

Table 4.4 *Objective Performance of GSM Speech Coder, with and without the LTP.*

Table 4.4 compares the objective performance of the GSM speech coder with and without the long-term predictor. With the LTP, there is a significant improvement in SNRSEG corresponding to the increase in subjective quality. The log-spectral distance is improved slightly. This has been achieved at the expense of increasing the coder data rate by 1800 bit/s.

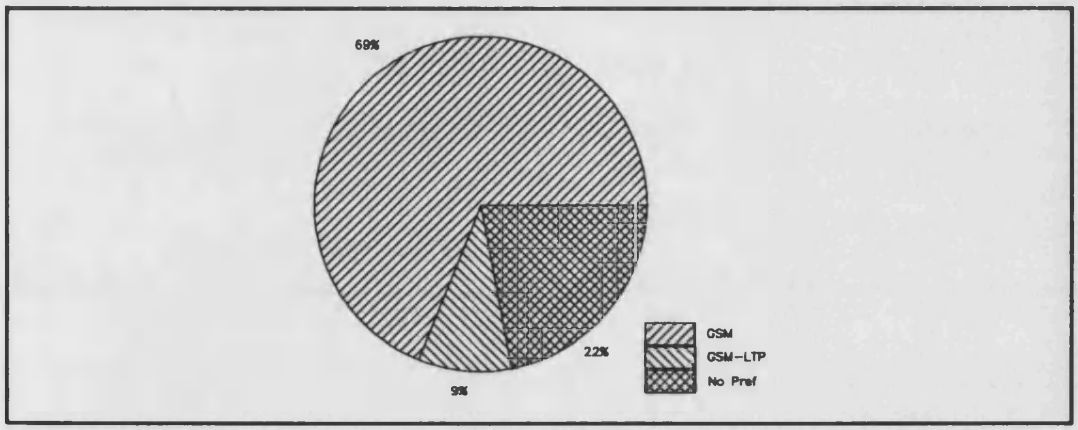

Figure 4.7 *Subjective Comparison of GSM Speech Coder with and without Inclusion of the LTP.*

Use of the LTP in this speech coder significantly improved the subjective quality. Results of subjective paired comparison of sentences produced by the normal GSM coder and the GSM coder without the LTP are shown in figure 4.7. Output produced by the normal coder was preferred in 69% of the tests compared to 9% preferring the output of the coder without the LTP. In informal listening, the normal coder with the LTP, differences between input and output speech could only be heard with concentrated listening through high-quality headphones. However an adverse affect of incorporation of the long-term predictor was the doubling of the simulation run time.

4.6.2 Noisy Channel Performance

This section describes an investigation into the GSM speech coders robustness to channel errors. In the mobile telephony environment, speech coders have to survive random bit errors along with impulsive error bursts. To minimise the problem of error bursts, transmitted bits are interleaved by the channel coder and in the case of GSM are transmitted over a number of time-slots. Thus any one error burst is likely to have

a lesser effect on a number of transmitted parameters than to have a major effect on any one parameter. To simulate noisy channel performance, transmitted parameters are recorded to disk files without interleaving, and random bit errors are introduced in these files using the mobile radio channel simulator, which is described in appendix A1.4.

Objective performance in transmission errors has been investigated for this speech coder with and without the inclusion of the long-term predictor. Figure 4.8 shows the resulting graphs of SNRSEG and log-spectral distance against bit error rate. It must be stressed that these results are for the raw coder, without any form of channel coding. Of interest is the greater degradation in SNRSEG of the coder with the LTP, both coders having very similar SNRSEG and log-spectral distance at error rates above 1%.

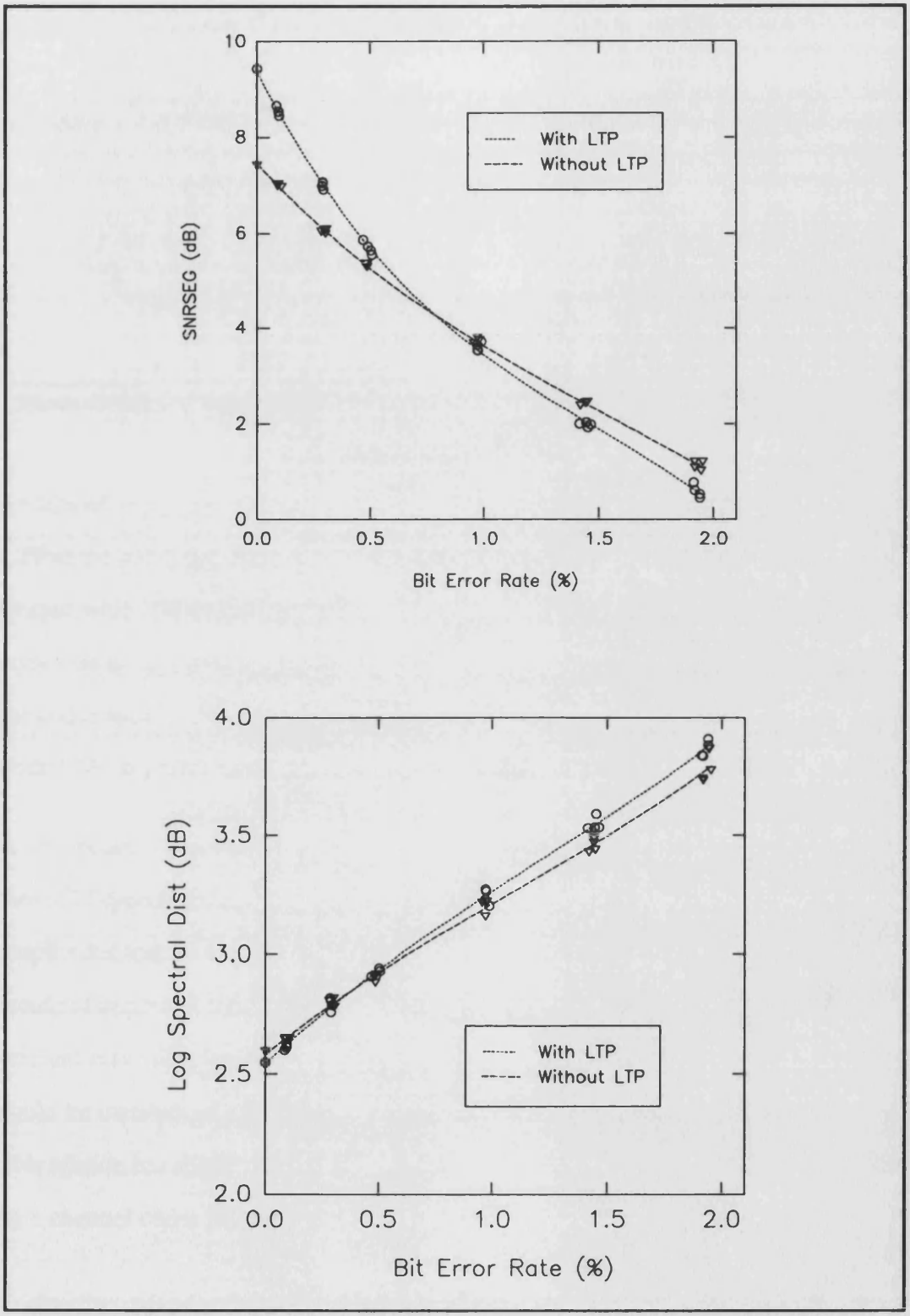

Figure 4.8 *Objective Performance of the GSM RPE Speech Coder with and without LTP, with Varying BER.*

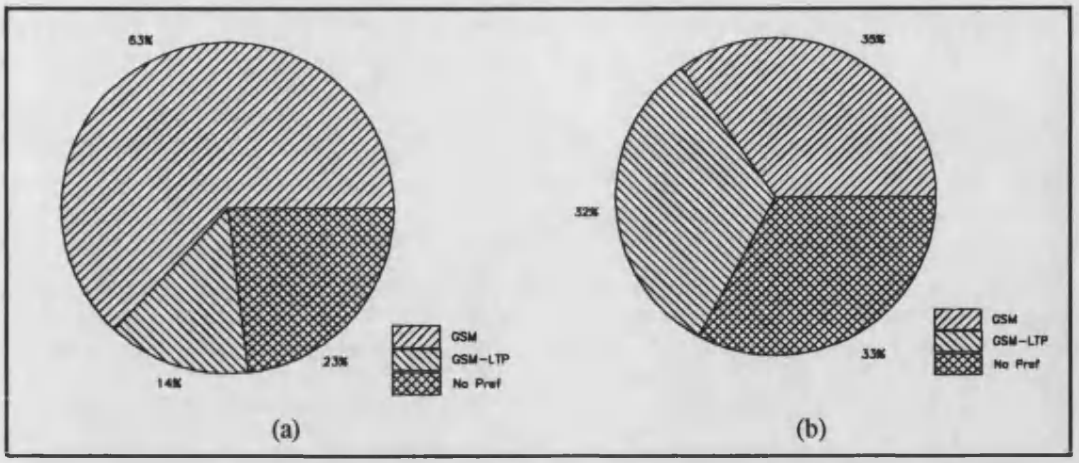

Figure 4.9 *Subjective Comparison of GSM Speech Coder with and without Inclusion of the LTP in noisy channel, (a) 1% BER, (b) 2% BER.*

Results of subjective paired comparison of sentences produced with and without the LTP in the presence of transmission errors at 1% and 2% BER are shown in figure 4.9. Output with LTP is still preferred at 1% BER (63% versus 14%), however when the error rate is increased to 2%, speech is still intelligible, however output produced by the coder with LTP is indistinguishable from that produced without (35% versus 32% with 33% no preference).

At this point, it should be noted that the original RPE coder without LTP submitted for the GSM speech codec trials used a greater number of bits to quantise to residual pulse amplitudes and the synthesis filter parameters, the overall bit rate was 14.77kbit/s, this would of improved the quality above the non-LTP coder described in this chapter. The original reasoning for the incorporation of a LTP was that equivalent sound quality could be maintained whilst the bit rate was reduced from 14.77 kbit/s to 13.0 kbit/s. This section has shown that this saving is only useful provided these freed bits are used by a channel coder for greater error protection.

4.6.3 Tandem Performance

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)
Single	9.44	2.55
Tandem	8.03	3.36

Table 4.5 *Objective Performance of GSM Speech Coder, Comparing single and tandem connection.*

One important requirement for the chosen speech coder was that intelligible conversation was possible when codecs are connected in tandem, as in mobile to mobile connection. Table 4.5 shows the objective performance of such a connection. Not surprisingly it is more distorted than the single connection case. Subjectively, conversation was still intelligible but distortion was more evident.

4.6.4 The Benefits of Pre-emphasis?

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)
With Preem.	9.44	2.55
Without Preem.	11.70	2.92

Table 4.6 *Objective performance of the GSM RPE speech coder with and without Preemphasis.*

Speech formants decrease in amplitude with increasing order. Traditionally in LP speech coding, the speech signal is often preemphasised, which raises the amplitudes of the higher formants and thus reduces the range of the speech spectrum. In addition a signals dynamic range is reduced which eases the fixed point implementation of LP coding. This section reports on what happens with this coder when the preemphasis and deemphasis functions are removed. Objective performance is shown in table 4.6. The SNRSEG has increased significantly, however the log-spectral distance has also increased, indicative of poorer spectral modelling. Informal listening showed the non-preemhasised output to be of higher quality indicating that the poorer spectral modelling is insignificant compared to the reduction of background noise.

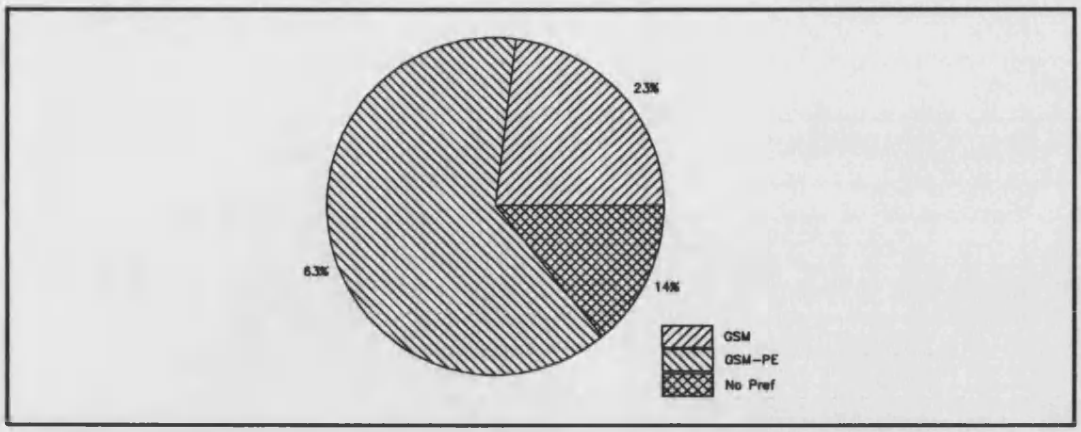

Figure 4.10 *Subjective Comparison of GSM Speech Coder with and without Preemphasis.*

Figure 4.10 shows the results of subjective paired comparison of sentences coded with and without the GSM preemphasis and deemphasis functions. Speech quality is perceived to be significantly better, without these functions (63% versus 23%).

A further investigation on the effects of preemphasis was to see the variation in the distribution of short-term filter reflection coefficients. This is shown in figure 4.11 for all eight reflection coefficients. They were measured over the same 53 seconds of speech consisting of 20 sentences from the Havard list of phonetically balanced sentences spoken by 10 males and 10 females. Each reflection coefficient was recorded over this time and the histograms constructed. They all have the same mean and deviation with and without preemphasis, except for r_1. Without preemphasis, this has considerable density near to -1. The companding characteristic of the conversion of reflection coefficients to Log-Area-Ratios is intended to spread out values in this area. Similar histograms of LARs are shown in figure 4.12. The conversion used was a linear piecewise conversion, as specified by GSM [10]. Notably this restricts the range of the LARs to ± 1.625. This has successfully spread out the many values at the lower end of the distribution for LAR_1. Although distributions are slightly different, the same filter parameter quantisation would be valid when preemphasis is not used.

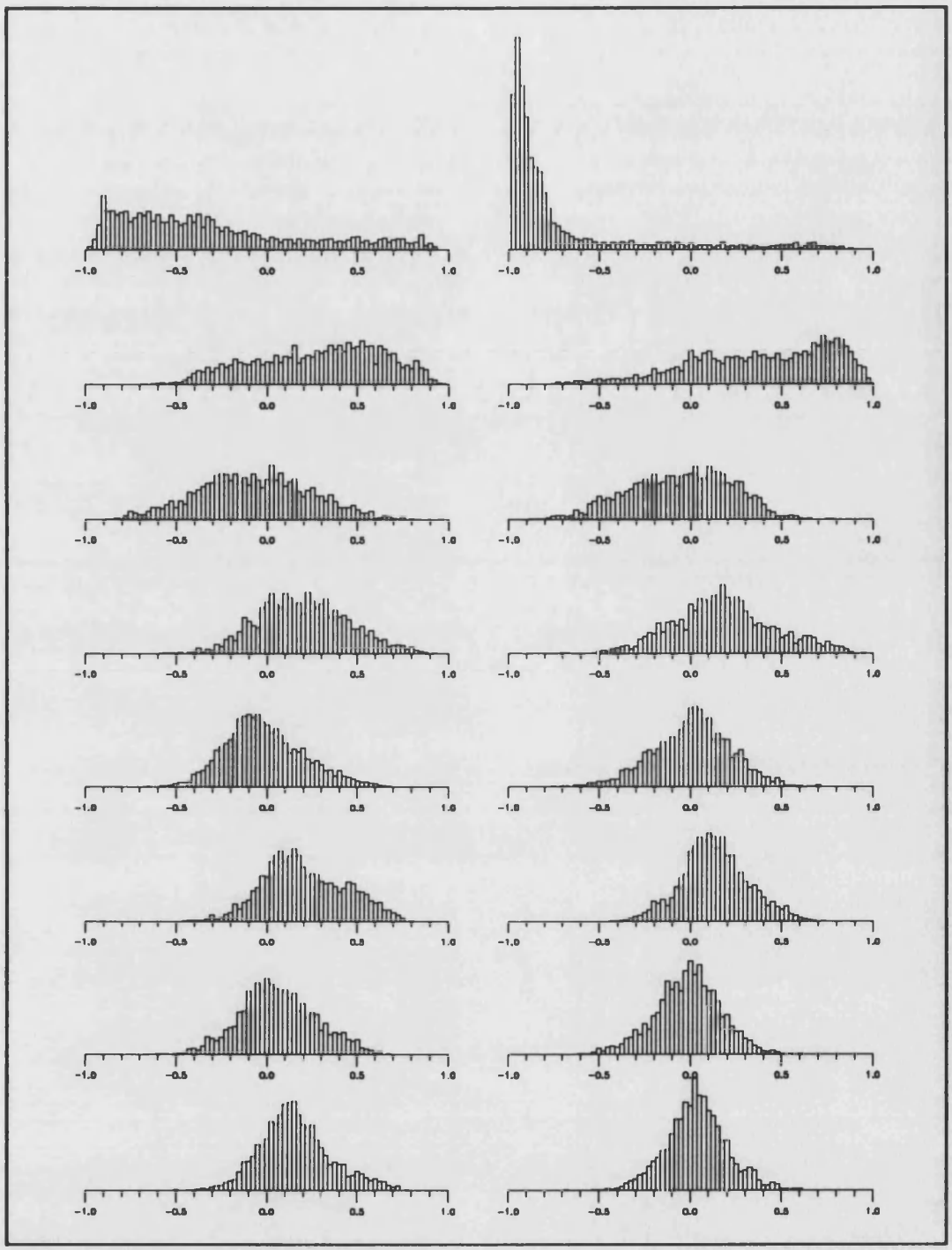

Figure 4.11 *Reflection Coefficient Histograms, left with Preemphasis, right without Preemphasis, top* r_1 *bottom* r_8.

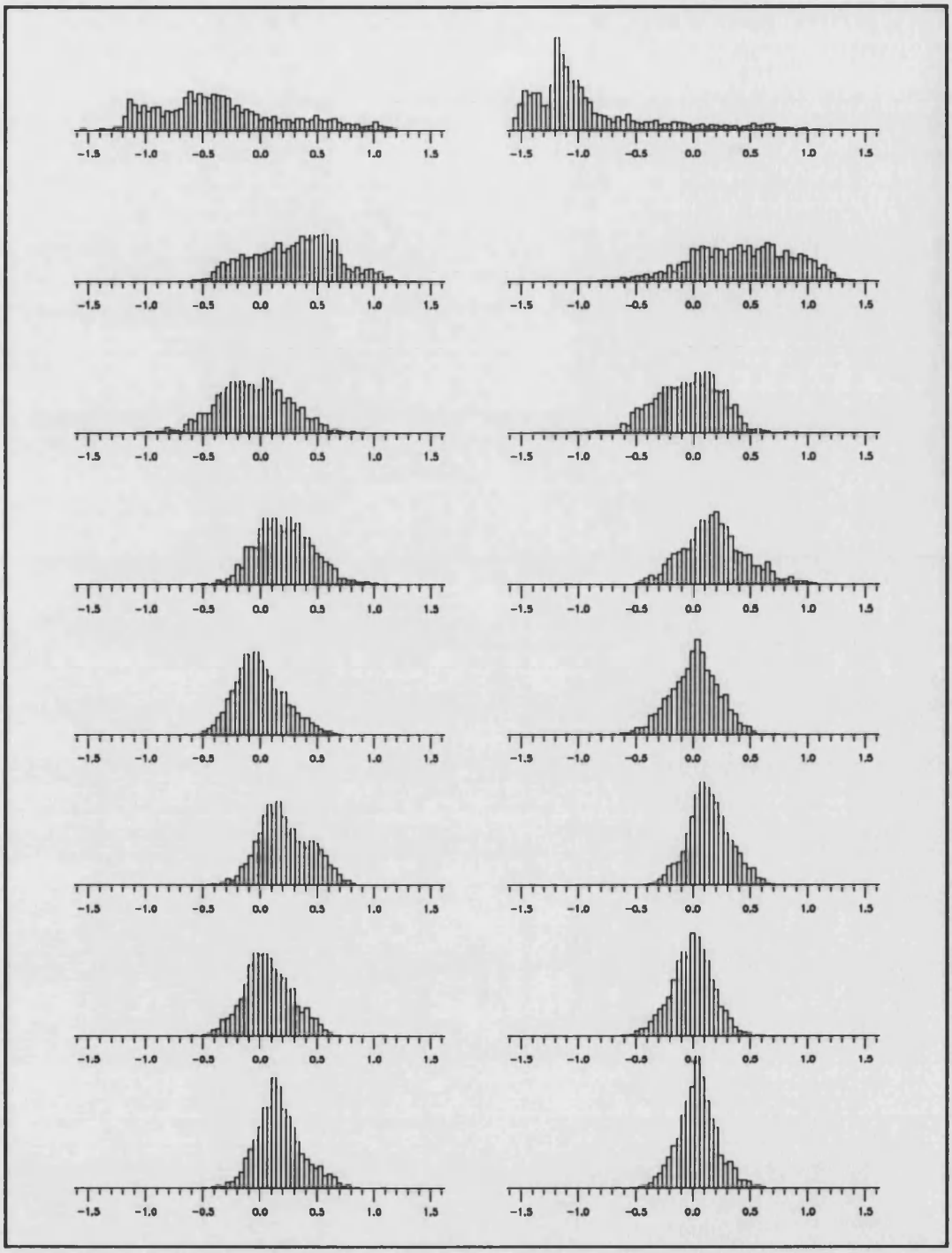

Figure 4.12 *Log-Area-Ratio Histograms, left with Preemphasis, right without Preemphasis, top* LAR_1 *bottom* LAR_8.

4.7 Summary

The RPELTP speech coder chosen by the GSM speech coding experts group for use in the pan-European telephone system has been implemented as a "C" simulation in non real time. Overall operation of this coder has been demonstrated figuratively by showing the passage of the utterance "Seven" through all the encoder and decoder stages. This showed how the coder removed and later recreated the redundancy from unvoiced, fricated and voiced sections of speech.

Inclusion of the LTP sections into both encoder and decoder was made switchable allowing the benefit of the LTP section to be assessed. With the LTP, coder waveforms showed less distortion. The SNRSEG and log-spectral distance measures showed better synthetic speech quality, this was particularly significant with the former measure. Paired comparison subjective testing showed a unanimous preference for synthetic speech utilising the LTP. Without the LTP, the coder distorted notably with periodic sounds such as tw*elve* and l*athe* etc.

The coders robustness to channel errors was tested by randomly corrupting bits transmitted from encoder to decoder. The raw coder was tested, without any form of channel coding. The coder was tested with error rates up to 2%, where speech was still intelligible. Both the complete coder and the coder without LTP sections were tested. Objectively, the superiority of the coder with the LTP at 0% BER is quickly lost and by 0.5% BER, their performance is well matched. Paired comparison subjective tests showed significant preference for the complete coder incorporating the LTP at 1% BER, however there was no preference between coders with and without LTP at 2% BER.

One of the initial speech coder requirements was that communication was still possible when mobile to mobile connection was made and two speech coders were connected in cascade. This was investigated and communication was still clear and intelligible, however, not surprisingly, distortion was more evident.

The final section of this chapter investigated the benefits of the preemphasis and deemphasis stages of the GSM speech coder. Without the functions, SNRSEG is significantly increased, indicative of significantly less output noise. However log-spectral distance is also significantly increased, indicative of poorer spectral modelling. Subjectively, there was a unanimous preference for synthetic speech produced without these functions. This showed the adverse effect of poorer spectral modelling was insignificant compared to the reduction of output noise. The distribution of both reflection coefficients and log-area ratios was studied with and without preemphasis concluding that the same parameter quantisation is valid when this function is not used.

Chapter 5

Low Complexity Self-Excited Vocoding

Chapter 5

Low Complexity Self-Excited Vocoding

This chapter develops a low complexity Self-Excited Vocoder and a low complexity CELP coder. Both are based upon the authors implementation of the GSM speech coder. The LTP section is modified firstly to form a Self-Exciting LTP (SE-LTP) and secondly a fixed codebook stage for the CELP coder. Sections 5.1 and 5.2 describe the SEV and CELP coder respectively. Section 5.3 extends these two coders further with the incorporation of a LTP. Finally, section 5.4 investigates SEV performance with multiple LTPs.

Objective results quoted in this chapter were obtained by coding/decoding a 53s speech test record containing 20 sentences from the Harvard list of phonetically balanced sentences [21] spoken by 10 male and 10 female speakers.

5.1 Low Complexity Self-Excited Vocoder (SEV)

Chapter 3 introduced the SEV, where the synthesis filter $1/A(z)$ excitation source is based upon its own previous output. This is achieved using a LTP without any input. The aim of this section was to develop a simple SEV based upon the authors implementation of the GSM speech coder.

The self-exciting LTP (SE-LTP) can only derive an excitation from its previous excitation history (the adaptive codebook), if a history exists. Thus to commence operation, a history must be established. Some studies of the SEV [43], used the actual LP analysis residual for the first few frames, before switching to SEV operation. This proved to demonstrate the technique, but required the decoder to be given information that was only available at the encoder. This is not possible in a practical SEV scheme. Later studies spoke of filling the history buffer with a zero-mean, unit variance, gaussian

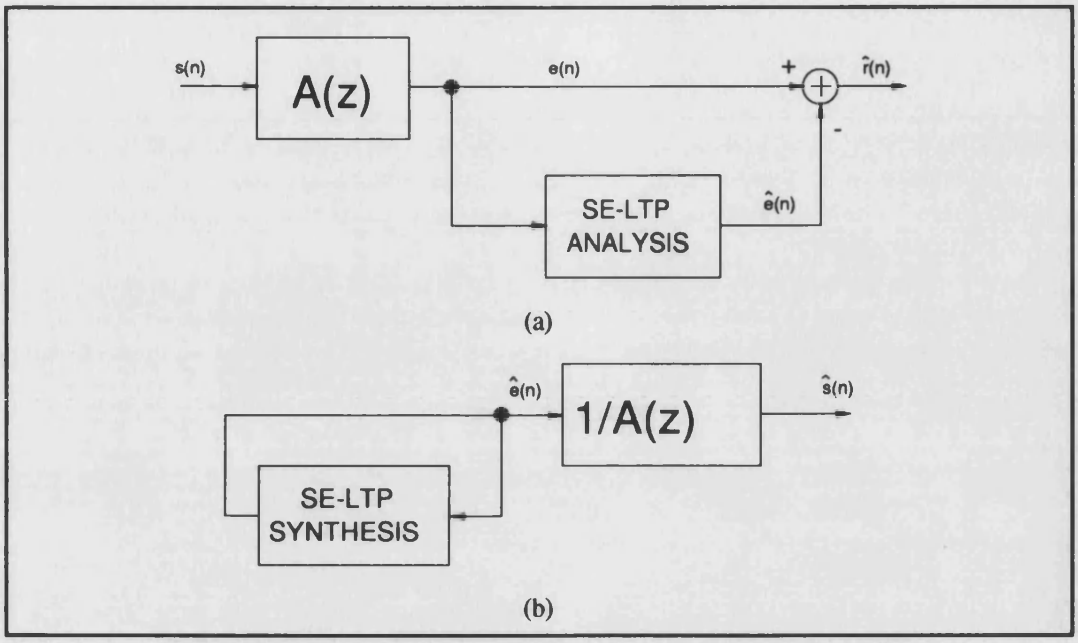

Figure 5.1 *Simple Self-Excited Speech Coder (a) encoder, (b) decoder.*

distributed random sequence at the start of the coding session [45]. For the purposes of this thesis, this approach is termed the pure SEV, and is the method used in this chapter.

The first modification to the GSM implementation, was code to initialise the LTP history with a zero mean, unit variance gaussian distributed sequence. This was achieved by the loading of a sequence from disk at coder startup, allowing the running of the coder with an easily changeable, pre-calculated initialisation sequence. This history was adapted by the normal LTP adaption process and the first SEV implemented had an adaptive codebook containing 81 adjacently spaced candidate excitation vectors of length 40 samples, as inherited from GSM. The initial preemphasis and final deemphasis stages of GSM were maintained, but were made switchable. The residual pulse encoding and decoding stages were discarded.

The block diagram of this simple SEV is depicted in figure 5.1, (a) the encoder and (b) the decoder. Analysis and synthesis filter parameters are determined over rectangular windowed frames of 160 samples of input speech $s(n)$ using autocorrelation and the

Leroux Gueguen method [30]. Input speech frames of 160 samples $s(n)$ are filtered by short-term analysis filter $A(z)$ giving short-term residual $e(n)$. SE-LTP analysis is performed over 40 sample subframes, giving the prediction signal $\hat{e}(n)$. This process is identical to that of the conventional LTP in analysis mode and is described by equations 3.49 to 3.55 in section 3.2.1. This speech coder has only one prediction stage, any signal unpredicted after this stage is lost.

Synthesis filter parameters, SE-LTP gain and delay are transmitted to the SEV decoder, depicted in figure 5.1(b). The SE-LTP synthesiser is initialised with an identical sequence, and reconstructs the prediction of the short-term residual signal $\hat{e}(n)$. This is input to the short-term synthesis filter $1/A(z)$ giving synthetic speech $\hat{s}(n)$.

The SEV encoder block diagram, figure 5.1(a), showed the SE-LTP adaption input to be the signal $e(n)$. In practice, $\hat{e}(n)$ is used, since this signal was available in both the encoder and decoder and enabled their adaptive codebooks to remain identical.

The LTP section from GSM quantised the predictor gain with 2 bits, into 4 distinct values, ranging from 0 to 1. Initial poor performance of first versions of this speech coder was traced to this gain quantisation, for three reasons. Firstly, experimentation showed significantly more than 2 bits were required for successful operation. Secondly, the gain range was dependant upon the rms amplitude of the codebook sequence and was not necessarily between these limits. Thirdly, better use of the codebook was made if gains could be either positive or negative. To overcome these problems, in these early experiments, the effects of gain quantisation were ignored and the value was passed from encoder to decoder unquantised.

This first low complexity SEV, designated "SEV1", used the 81 element codebook, inherited from GSM. Subjectively, the synthetic speech was intelligible, but of poor quality. Processing without the preemphasis and deemphasis stages did not change the nature of the reproduced speech, but resulted in a slightly less noisy, higher quality

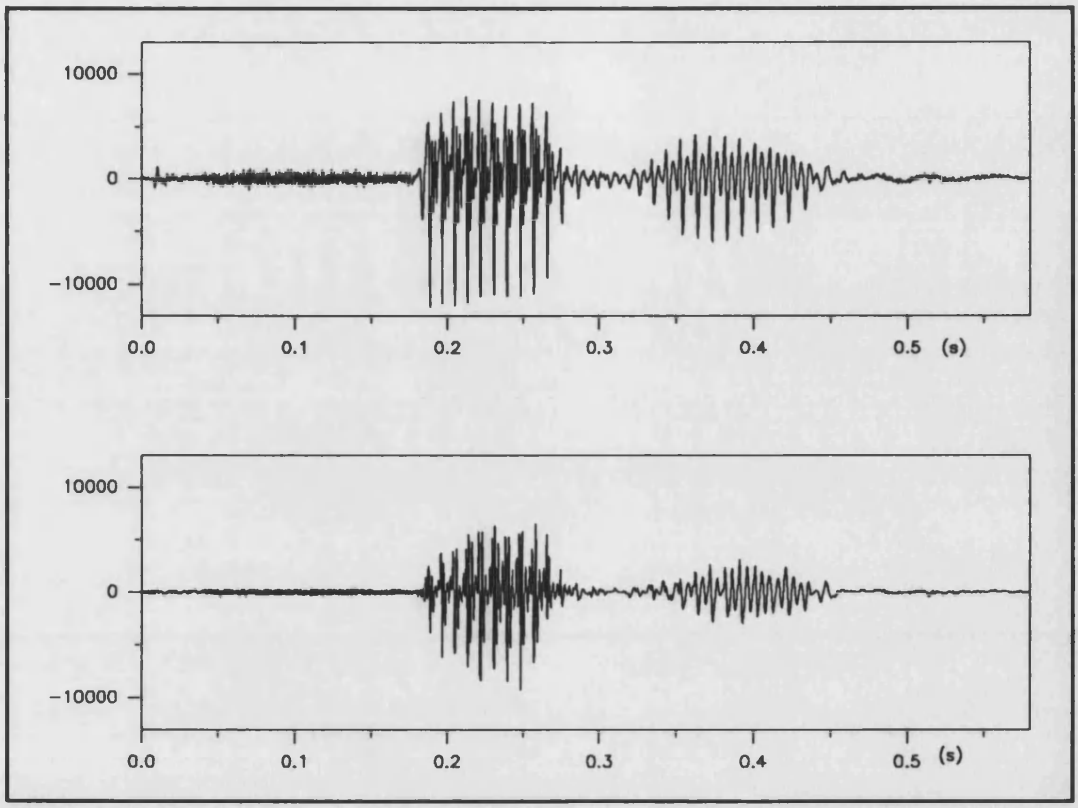

Figure 5.2 *Performance of the 81 Vector Codebook Simple SEV, the Input shown in the Top Trace is the Utterance "Seven" by a Male Speaker, Bottom Trace is the Coder Output.*

output. An example of this coder output is shown in figure 5.2, the top trace showing the input speech, in this case ("Seven"), and the bottom trace showing SEV1 output. (This output was obtained without preemphasis and deemphasis functions.)

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)
SEV1	1.78 (1.14)	3.85 (3.54)
SEV2	2.26 (1.45)	3.65 (3.40)

Table 5.1 *Performance of SEV Coders SEV1,2, Values in Parenthesis Correspond to Coder including Preemphasis and Deemphasis Stages.*

The objective results for this coder SEV1 are given in table 5.1. They are the average of four runs coding/decoding the test record, each time with a different initialisation sequence. Objective speech quality of this very simple coder is poor, with low SNRSEG

and high spectral distance measures. The next experiment was to increase the codebook size giving 256 possible excitation vectors. This was coder "SEV2". Objective measures, also shown in table 5.1 are a notable improvement but are still very low for a practical speech coding scheme. Although still very low, the SNRSEG is increased without the preemphasis and deemphasis stages which corresponds with a noticeable improvement in subjective quality. However the Log-Spectral Distance, a frequency domain objective measure, is worsened, indicative of poorer short-term modelling of the speech waveform. However this adverse effect was subjectively insignificant compared to the reduction of output noise shown by the increased SNRSEG measure.

5.2 Low Complexity CELP Coder

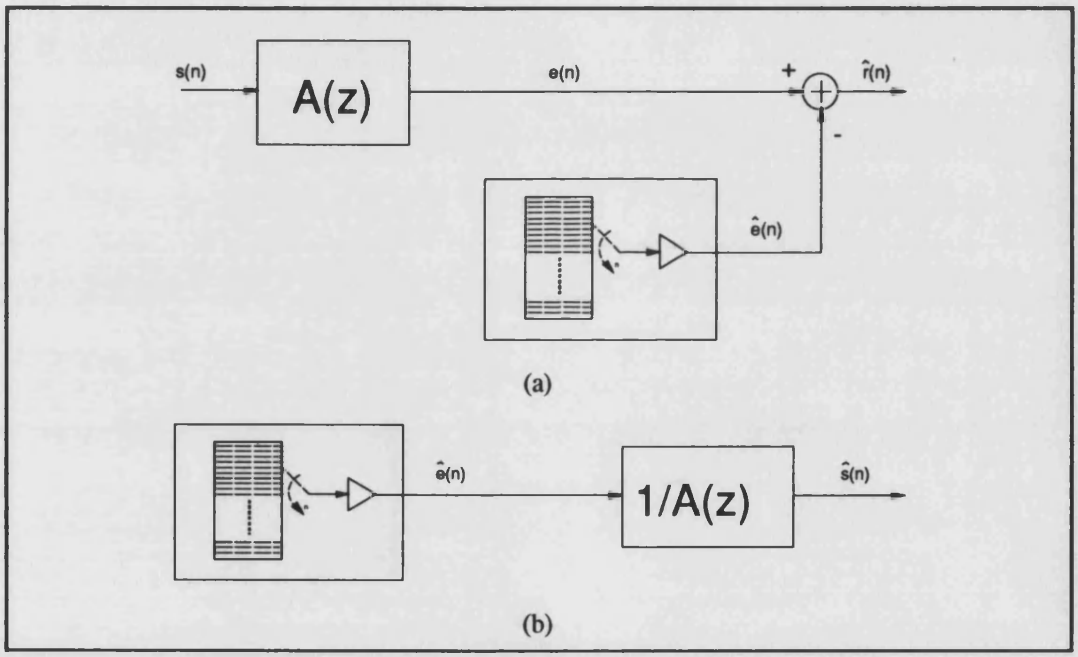

Figure 5.3 *Simple CELP Speech Coder (a) encoder, (b) decoder.*

The previous section developed two simple SEVs. This section implements two similar CELP coders to assess the benefit (if any) of the adaptive codebook over a fixed codebook. The simple CELP coder was a modification of the simple SEV program and was obtained by simply removing the codebook adaption sections. This ensured that the codebook contents were unchanged during coder operation. The block diagram of

the CELP coder is shown in figure 5.3, (a) showing the encoder, (b) showing the decoder. Operation is very similar to that of the SEV, with a fixed codebook stage in place of an adaptive codebook stage.

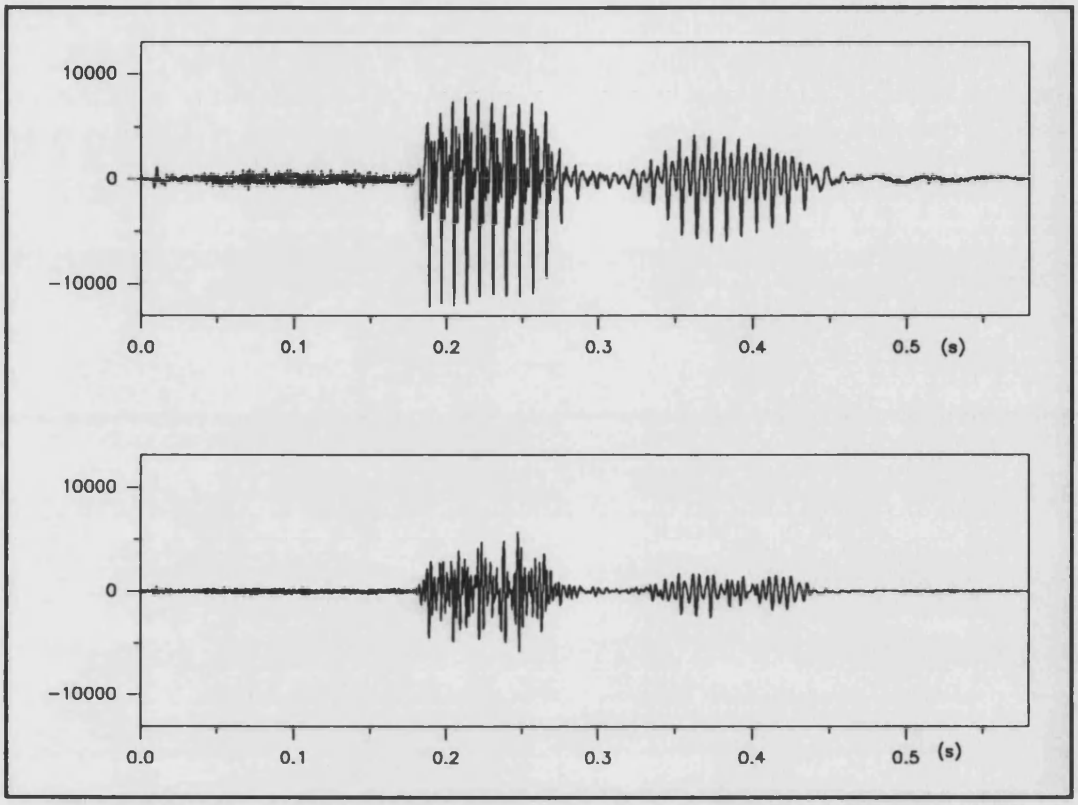

Figure 5.4 *Performance of this Simple CELP Coder, the Input shown in the Top Trace is the Utterance "Seven" by a Male Speaker, Bottom Trace is the Coder Output.*

"CELP1" was the first CELP coder and consisted of an 81 element codebook. The performance of this coder over the utterance "Seven" by a male speaker is shown in figure 5.4, which shows both input with output. This is notably inferior to the SEV output. Reproduction of pitch periodicity is poor and this is particularly noticeable in the centre fricated segment and the end voiced segment. The amplitude of the entire waveform section is low. This is directly attributable the a very poor match of codebook excitation vectors with the analysis filter residual subframes. This can be explained with reference to equation 3.53 where the predictor gain is calculated. The numerator

is the result of cross correlation between codebook vector and analysis filter residual. If this is low, indicative of a bad codebook match, the resulting gain and hence speech power will be low.

Subjectively, the synthetic speech had a very rough, gravelly sound, which although intelligible was significantly inferior to that of the SEV. Processing without the preemphasis and deemphasis stages did not change the nature of the reproduced speech, but again resulted in a slightly less noisy output. The objective results for this coder CELP1 are given in table 5.2.

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)
CELP1	1.04 (0.88)	4.07 (3.70)
CELP2	1.33 (1.14)	3.77 (3.41)
CELP3	1.28 (0.97)	3.79 (3.45)

Table 5.2 *Performance of CELP Coders CELP1,2,3, Values in Parenthesis Correspond to Coder including Preemphasis and Deemphasis Stages.*

This limitation in performance of this simple vocoder may have been the result of a very small number of codebook vectors. The next experiment was to increase the codebook size giving 256 possible excitation vectors. This was coder "CELP2". When subjectively compared alongside CELP1 the performance was only slightly superior and the gravelly characteristic was still very present. The results of objective tests, also shown in table 5.2 are a significant improvement but are still extremely low for a practical speech coding scheme.

A further experiment was increasing the spacing between adjacent codebook vectors to two samples, ie. neighbouring candidate vectors overlap for all but two samples. This required lengthening of the codebook buffer by 256 samples. It is reported [23] that codebooks of this nature have performance as high as codebooks containing fully

independent vectors. This was coder "CELP3". Objective performance of this coder was slightly inferior to CELP2 but not significantly enough to draw any firm conclusions. In fact, informal listening could not distinguish between outputs.

Increasing the codebook size improves the synthetic speech quality but nowhere near enough for a practical speech coding scheme. The limitation is now in the coders inability to reproduce pitch periodicity. This is the subject of the next section, where performance will be enhanced by the incorporation of a LTP.

5.3 Addition of a Long-Term Predictor

The next series of experiments focussed on adding a conventional LTP to the simple speech coders already described. The LTP used was based on that used in the GSM speech coder and operated with delays in the range of 5-20ms. The LTP was modified from GSM such that the gain value was passed from encoder to decoder unquantised.

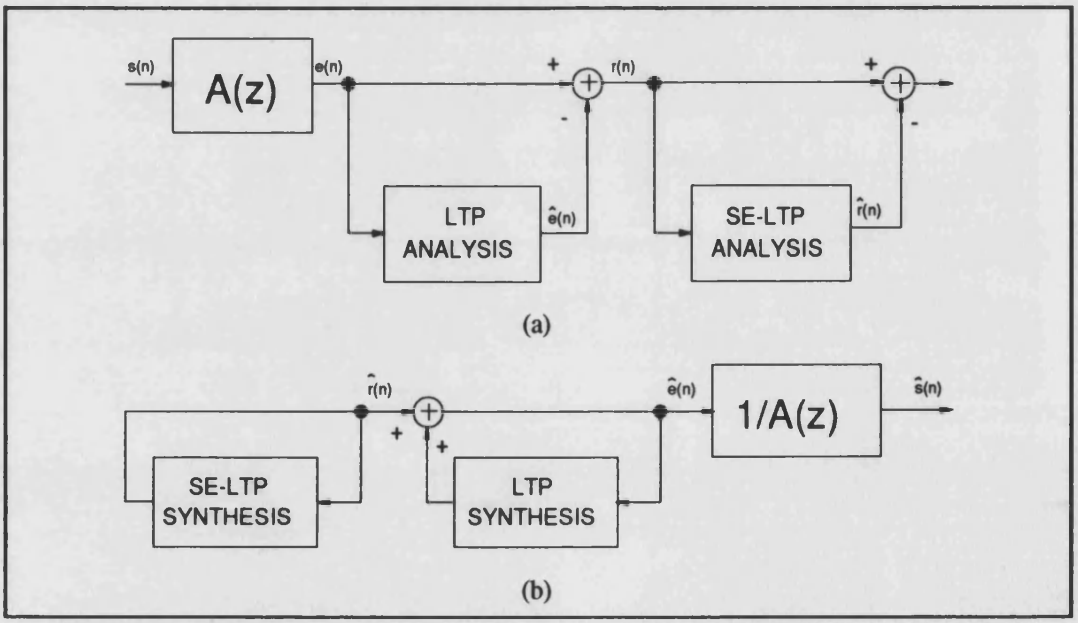

Figure 5.5 *Self-Excited Speech Coder Incorporating a Long-Term Predictor, (a) encoder, (b) decoder.*

Figure 5.5 shows the incorporation of the LTP into the simple SEV in (a) the encoder and (b) the decoder. After short-term analysis filtering, the signal $e(n)$ is long-term analysis filtered, leaving signal $r(n)$. The SE-LTP output $\hat{r}(n)$ predicts this short and long-term residual. Any signal unpredicted after these two predictor stages is lost.

The decoder is the inverse of the encoder, firstly recreating the short and long-term residual $\hat{r}(n)$ in the SE-LTP synthesis stage. The short-term residual is then recreated after the LTP synthesis stage and finally synthetic speech $\hat{s}(n)$ is output after the short-term synthesis filter $1/A(z)$ stage.

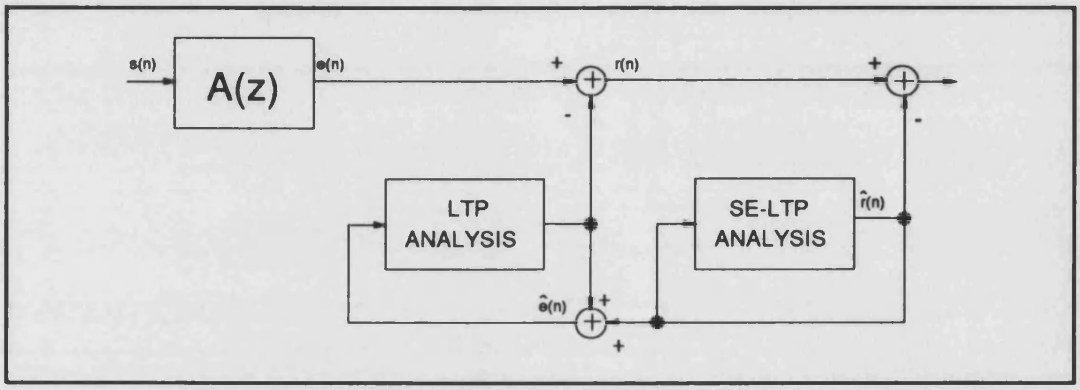

Figure 5.6 *Actual Signal Flow within the SEV incorporating a LTP.*

LTP operation in analysis mode, has already been described in section 3.2.1. The important point that must be raised from figure 5.6 is the mechanism for the updating of the adaptive codebook. Theoretically, this should be a history of the actual short-term residual $e(n)$, this signal is only available at the encoder and not the decoder. For satisfactory operation, the LTP history in both encoder and decoder must be identical. This can only be achieved if the LTP history is based upon the synthetic short-term residual $\hat{e}(n)$. Thus in practice the signal flow within the encoder is that depicted in figure 5.6.

The LTP was added to coder SEV2 to give coder SEV4. A LTP was also added to coder CELP2 to give CELP4. Both the SE-LTP stage of SEV4 and the codebook stage of CELP4 had 295 elements giving them 256 adjacently spaced candidate, 40 sample

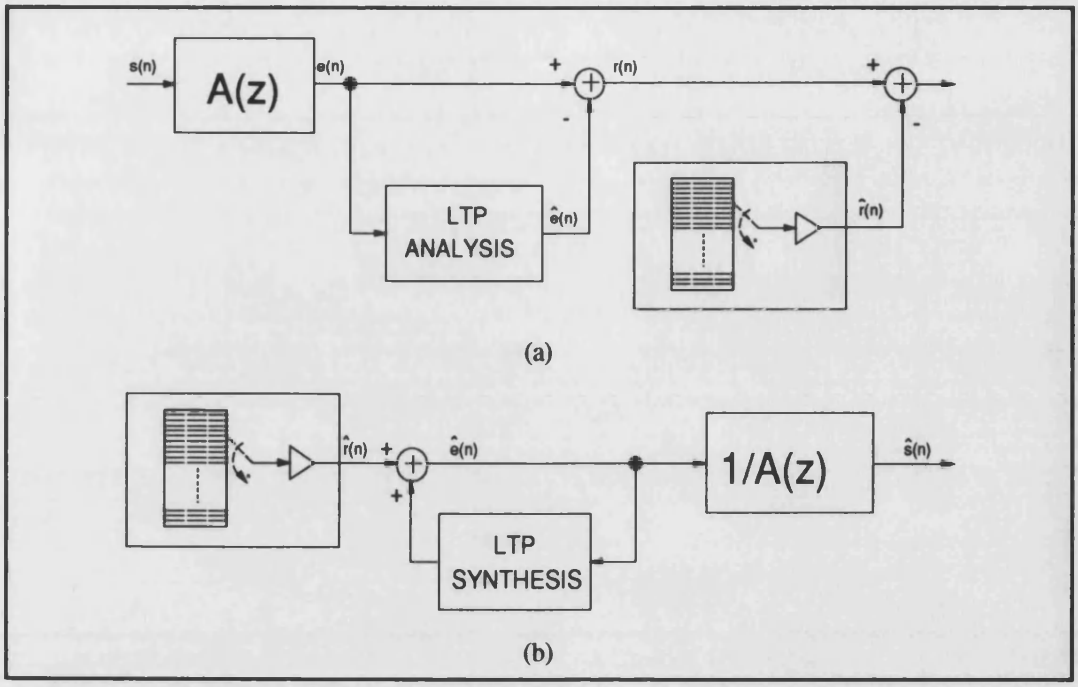

Figure 5.7 *CELP Speech Coder Incorporating a Long-Term Predictor, (a) encoder, (b) decoder.*

excitation vectors. The coder CELP4 was again a modification of its SEV counterpart, without the adaption of the first codebook stage. Coder CELP4 is depicted in figure 5.7.

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)
SEV4	4.05 (2.87)	3.01 (2.85)
CELP4	4.03 (3.23)	2.95 (2.81)

Table 5.3 *Performance of CELP4 and SEV4 Coders, Values in Parenthesis Correspond to Coder with Preemphasis and Deemphasis Stages.*

The objective results for both coders are given in table 5.3. These are a significant improvement upon their without LTP counterparts, the SNRSEG for the CELP coder having tripled, and for the SEV having doubled along with significant improvements in the Spectral Distance. With the LTP, the objective performance of both coders is very similar, there is no longer a notable superiority of the SEV over the CELP coder.

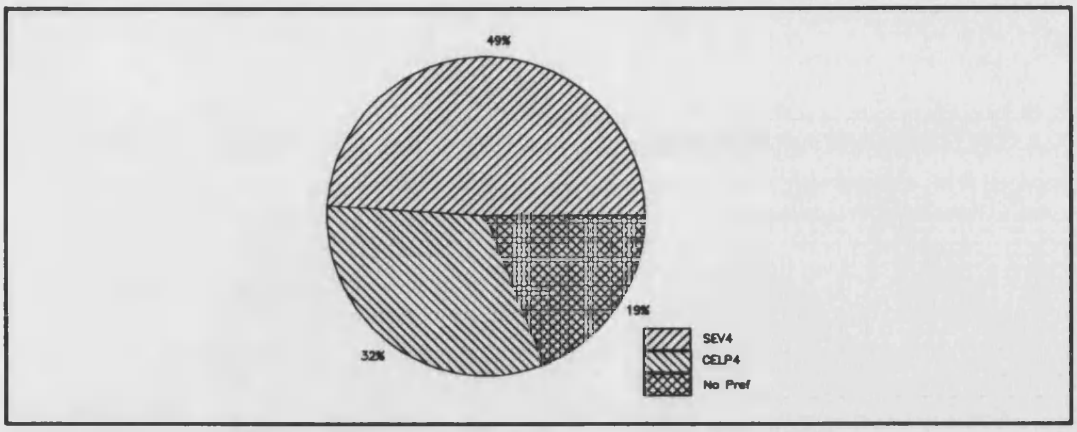

Figure 5.8 *Subjective Comparison of SEV4 and CELP4.*

Addition of the conventional LTP has dramatically improved the coders subjective performance, the output is significantly clearer and of higher quality. The subjective quality of both coders SEV4 and CELP4 is close. The subjective preference of both techniques was tested by paired comparison of both male and female synthetic sentences listened to by 50 listeners, the results are shown in figure 5.8. Despite the very similar objective results, there remains a significant preference for the SEV output (49% versus 32%). Both coders synthesised male voices considerably better than female voices. Although this is common in LP speech coding, it is not helped by this particular LTP, which is only able to model pitch periods down to 5ms.

5.4 Multiple Long-Term Predictors

Such a significant improvement was obtained with the addition of a single LTP, that the question was raised, could this improvement be extended with the addition of even more. Early experiments with multiple copies of LTP code lines identified the correct manner of sequentially optimising LTP gain and delay values, and the updating all the adaptive codebooks. Superior performance was obtained by first optimising parameters from the last LTP of the chain, followed by the next last etc. Adaption of codebooks was the reverse order with each successive LTP requiring the output from the previous for its own codebook adaption.

Eventually one multiple LTP module was written, which through the use of multidimensional arrays, enabled a SEV to incorporate anything between 2 and 11 LTPs. The required number was entered on the program command line. This SEV was termed "SEV5" and again passed gain values from encoder to decoder unquantised.

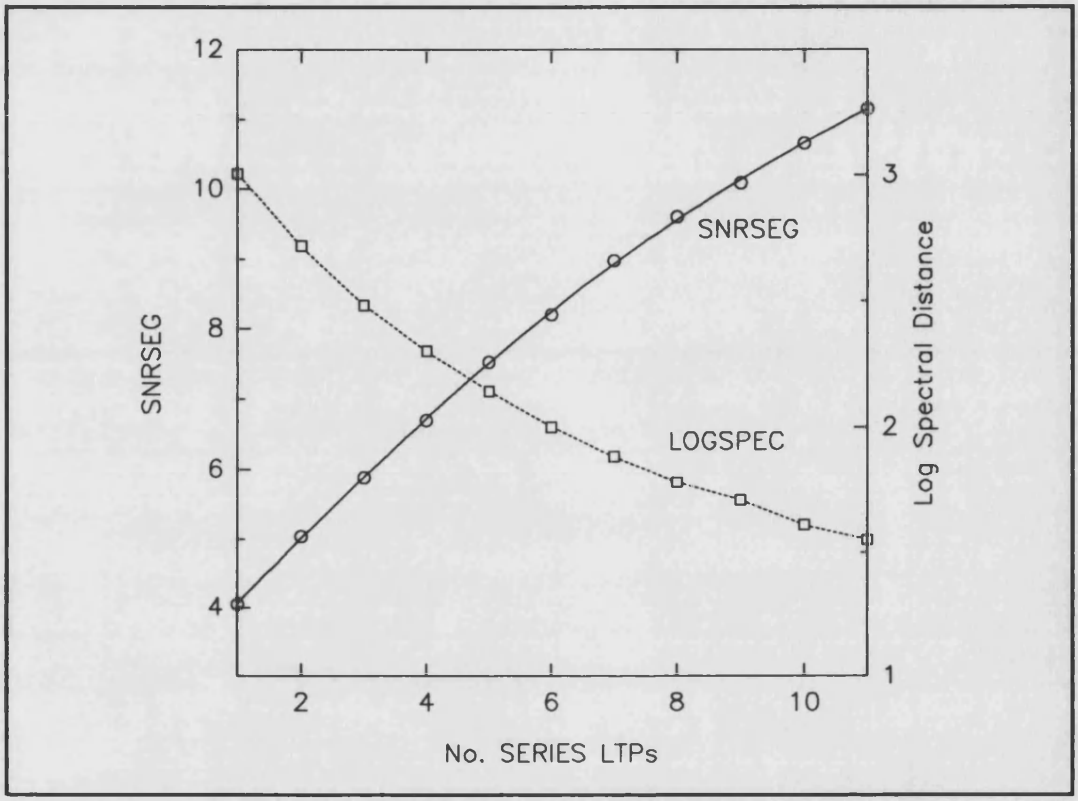

Figure 5.9 *Objective Performance of SEV5 with Multiple LTPs.*

Figure 5.9 shows the objective performance of a SEV with inclusion of from 1 to 11 LTPs. Performance increases rapidly as the number of LTPs is increased and excellent, very high SNRSEG scores and very low spectral distances are achieved. Subjective quality was also excellent. This is illustrated in figure 5.10, which compares the subjective preference of SEV5 with 11 LTPs with the GSM speech coder. There is significant preference for the multiple LTP SEV (44% versus 26%).

This experiment demonstrates the value of LTPs in improving speech quality in speech coding. However, it must be stressed, this is not realistic speech coding scheme as many LTPs require a high bit rate and have little tolerance to channel errors.

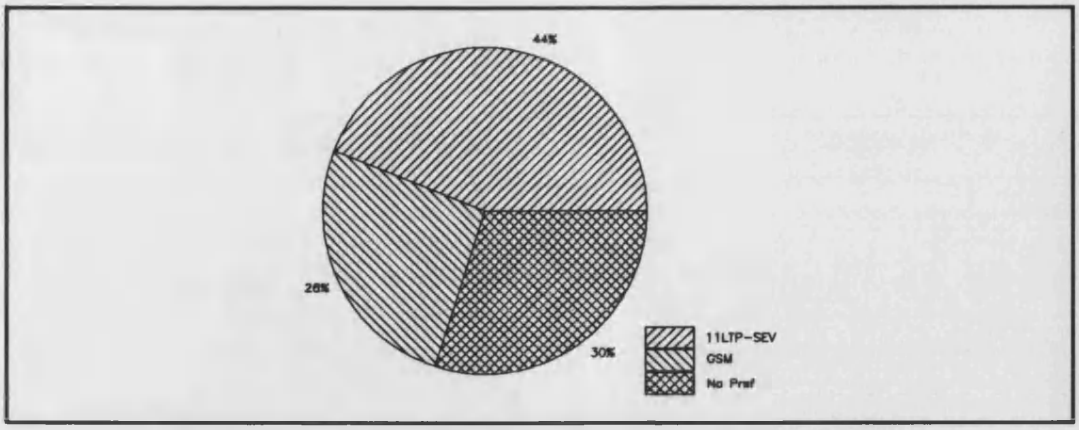

Figure 5.10 *Subjective Comparison of SEV5 (with 11 LTPs) and GSM.*

5.5 Summary

This chapter has developed simple low complexity SEVs and CELP coders, based upon the authors implementation of the GSM speech coder. The first coder developed was a Pure SEV, which demonstrated the operation of the SE-LTP without any input, but in this very simple state had a poor quality output. Synthetic speech quality was improved by increasing the length of the adaptive codebook to give 256 candidate excitation vectors as opposed to the 81 inherited from GSM. For reference, a simple low complexity CELP coder was also developed, having an equal sized fixed codebook as the SEV adaptive codebook. This had notably inferior performance both objectively and subjectively, with the synthetic speech having an annoying rough, gravelly nature. Again performance of this coder was increased slightly by extending the fixed codebook to 256 vectors. There was no significant difference in the coders performance when the spacing of adjacent codebook candidate sequences was increased to 2.

With all the coders, the subjective quality and the SNRSEG were significantly improved without the preemphasis and deemphasis functions of the GSM coder. conversely the log-spectral distance measure was worsened, indicative of poorer spectral modelling. This adverse effect was subjectively insignificant, compared to the reduction of output noise shown by the increased SNRSEG measure.

Addition of a LTP to both types of coder gave a dramatic improvement, both SEV and CELP coder had near equal objective performance. However the SEV had significantly superior subjective quality as confirmed by a paired comparison listening test.

The final section of this chapter experimented by incorporating many LTPs into a SEV, results were obtained from between 1 and 11. Very good speech quality was achieved both objectively and subjectively, with a paired comparison subjective test showing quality superior to that of the GSM speech coder with an 11 LTP SEV. Whilst not suitable for a practical speech coding scheme, multiple LTPs serve to illustrate the value of the LTP in improving synthetic speech quality.

Chapter 6

Analysis by Synthesis Self-Excited Vocoding

Chapter 6

Analysis-by-Synthesis Self-Excited Vocoding

The previous chapter reported experiments with low complexity, non analysis-by-synthesis, CELP coders and SEVs. Performance of these speech coders with realistic numbers of LTPs was limited, however the chapter did highlight the effectiveness of the SEV technique over the CELP technique, ie performance of a codebook is superior when it is based upon the synthesis filter excitation history rather than fixed random sequences. Output SNRSEGs of 4dB were obtained which is far too low for a practical system. This chapter expands upon the techniques of the previous chapter and introduces high complexity, analysis-by-synthesis techniques, coupled with perceptual error weighting, to utilise the masking properties of the human ear, to determine optimum excitation vectors. In addition, coders developed within this chapter will be fully quantised enabling the assessment of their robustness to channel errors.

The initial experiments of chapters 4 and 5 prompted the development of a speech coder module which could be easily converted into a full speech coder program. This is described in detail in Appendix A1.2. All the speech coders described in this chapter were developed from this code section.

This chapter is divided into 6 sections. Section 6.1 discusses the conditions a speech coding scheme must survive in the mobile telephony environment, section 6.2 develops a number of reference CELP coders enabling performance comparison with the SEVs developed later in the chapter. Their performance is studied both with and without LTPs and in the presence of channel errors. In addition, codebooks are made up of both gaussian sequences and ternary sequences. Section 6.3 develops the analysis-by-synthesis Pure SEV, showing good speech quality is achieved, however with a total lack of error robustness. Section 6.4 develops the Partial Fixed Codebook (PFC) SEV and the Separate Fixed Codebook (SFC) SEV which allow some tolerance

to channel errors. Section 6.5 introduces two new codebook adaption schemes which further improve coder error robustness, at the expense of some clear channel performance. The final section, 6.6, studies the effect of changing the gaussian initialisation sequence of the SEV to a ternary one.

Objective results quoted within this chapter were taken using a 53 second long speech record containing 20 sentences from the Harvard list of phonetically balanced sentences [21] spoken by 10 males and 10 females. The same record was used in the objective results of the previous chapters. Considerable attention is paid to the noisy channel conditions in which a speech coder must operate. Tests are conducted on channels with random bit error rates up to 1 or 2% and comparison in performance is made between different speech coders. Much use is also made of paired comparison subjective testing to compare outputs from different speech coders. Results quoted are from 50 listeners hearing two paired comparisons, one comprising a male speaker, and one a female. The subjective testing procedure is described in appendix 4.

6.1 The Mobile Telephony Environment

When a vocoder is to operate in the mobile telephone environment, satisfactory performance must be maintained over noisy channels. This means, speech quality must degrade gracefully in random errors, or short error bursts. Alternatively, recovery must be rapid from longer breaks in transmission. These could result from fading, channel stealing (where the voice channel is used to transmit important system information) or discontinuous transmission (where a mobile telephone only transmits whilst the user is speaking.) A practical speech coder must have inherent robustness, where an error in one particular bit has a short lived effect on the decoded speech. The effects of channel errors must propagate as little as possible.

The distortion encountered in the mobile telephony environment consists of random bit errors and impulsive error bursts. To minimise the problem of error bursts,

transmitted bits are normally interleaved by the channel coder and transmitted over a number of time-slots. Thus any one short error burst is likely to have a lesser effect on a number of transmitted parameters than to have a major effect on any one parameter. To simulate noisy channel performance, transmitted parameters are recorded to disk files without interleaving, and random bit errors are introduced in these files using the mobile radio channel simulator, which is described in appendix A1.4.

There is an important distinction between inherent error robustness and error protection coding. Inherent error robustness is the ability of a raw coder, without any associated error protection coding, to withstand channel errors with as little degradation to the synthetic speech as possible. This thesis studies the inherent error robustness of SEVs.

Alternatively, with error protection coding, parameters are transmitted with redundancy, allowing bit errors with serious effects to be detected and either corrected or other measures taken, such as re-using a previous frame value [4]. Should transmission errors become too severe, more drastic action must be taken, possibly the repeating of the previous speech frame, with a slight attenuation. This error protection coding is outside the scope of this thesis and is the major subject for continuation of this research.

6.2 Reference CELP Coders

The Code Excited Linear Predictive (CELP) speech vocoder is now well established, a version has recently been adopted as the US Federal standard for low bit rate speech transmission [12]. For this reason, SEV results will be regularly compared with those of the CELP coder. Using the speech coder module described in appendix A1.2, three reference CELP coders have been constructed, and these have been tested with both gaussian random codebooks and ternary codebooks. The ternary codebook elements have only three distinct values, these being -1, 0, +1 [54] and this type of codebook is gaining wide acceptance.

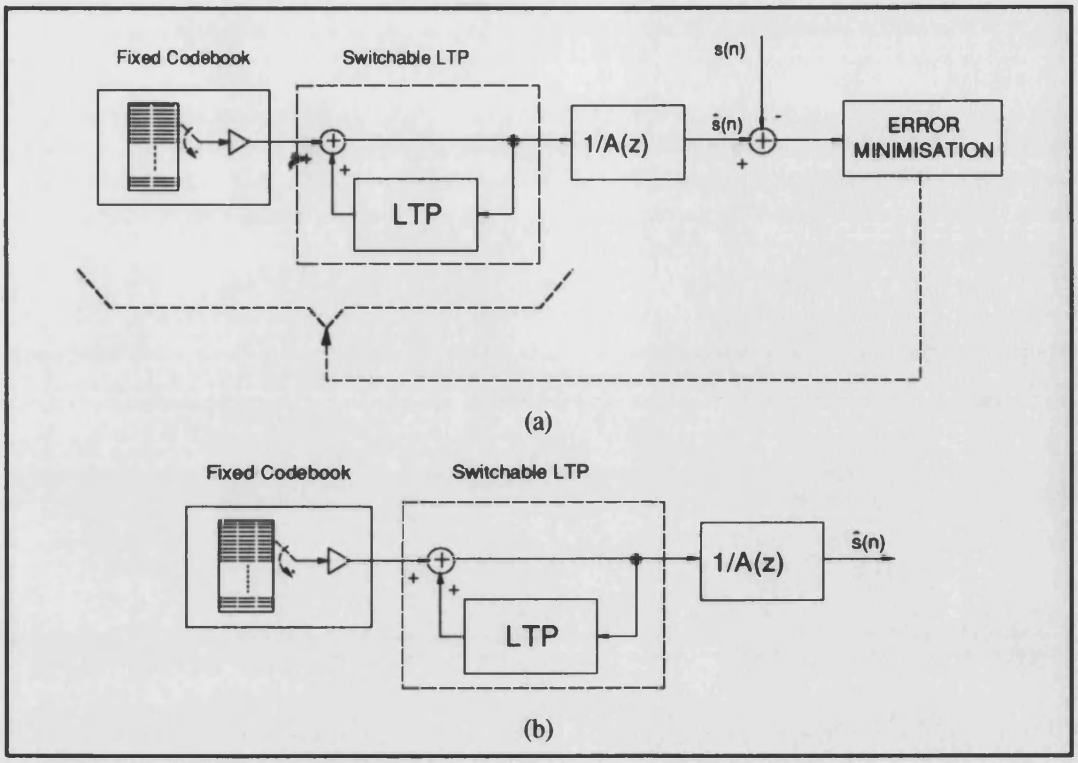

Figure 6.1 *Analysis-by-Synthesis CELP Coder with Switchable LTP, (a) Encoder, (b) Decoder.*

The reference CELP speech coder is depicted in figure 6.1. The codebook consists of a stochastic sequence containing overlapping excitation vectors. The chosen codebook vector is amplified/attenuated before input to the next stage. If the LTP is enabled the pitch is modelled as described in section 3.2.2 and the LTP output is used to excite the synthesis filter $1/A(z)$ giving output speech. If the LTP is not enabled, the output from the fixed codebook stage is used to excite the synthesis filter $1/A(z)$. The box labelled "error minimisation" refers to the determination of optimum codebook vector, optimum LTP delay and the corresponding gains to minimise the error between synthetic and original speech. In practice this is performed by optimising LTP parameters before fixed codebook parameters. With perceptual error weighting, this error minimisation occurs after the difference signal has been passed through a perceptual error weighting filter $A(z)/A(bz)$, where $b = 0.8$. The error weighting can be enabled and disabled by setting a switch on the program command line.

The synthetic speech output from the reference CELP coder, incorporating the LTP, is given by

$$\hat{s}(n) = \gamma v_l(n - d_l) * h(n) + \phi v_F(n - d_F) * h(n) + \hat{m}(n) \qquad 6.1$$

where v_l, d_l and γ are the LTP adaptive codebook buffer, delay and gain respectively and v_F, d_F and ϕ are the fixed codebook buffer, index and gain respectively. $h(n)$ is the impulse response of the synthesis filter $1/A(z)$ and $\hat{m}(n)$ is the contribution due to its memory. Minimisation of the mean squared error gives

$$E = \sum_{n=0}^{N-1} \tilde{s}^2 - \gamma \sum_{n=0}^{N-1} \tilde{s}(n)\,[v_l(n - d_l) * h(n)] - \phi \sum_{n=0}^{N-1} \tilde{s}(n)\,[v_F(n - d_F) * h(n)] \qquad 6.2$$

where N is the analysis subframe length. When the LTP is not incorporated, the contribution due to the LTP is ignored in the above equations and the fixed codebook stage parameters ϕ and d_F optimised accordingly. Alternatively, when the LTP is incorporated, joint optimisation of the mean squared error for all possible combinations of d_F and d_l would present an excessive computational task. Instead they are optimised sequentially, with LTP parameters γ and d_l optimised first, followed by fixed codebook parameters ϕ and d_F.

Three variants of reference CELP coder have been constructed, these are termed CELP1, CELP2 and CELP3. They are all identical apart from the number and spacing of the fixed codebook vectors. Since most of the SEV work will consist of codebooks with 256 vectors, CELP1 and CELP2 both have codebooks with 256 vectors, CELP1 has adjacent vectors overlapping for all but one sample, and CELP2 for all but two samples. Increasing the spacing to two samples has been shown to give the codebook performance equivalent to a codebook with fully independent codebook vectors [23], and this is accepted practice in CELP coding. CELP3 has 512 excitation vectors spaced by two

samples and hence a higher bit rate. The speech coders were constructed such that the fixed codebook sequence was loaded at the start of the coding session. This allowed either a gaussian or a ternary sequence to be loaded.

Linear Predictive analysis is performed on non-preemphasised, non-overlapping rectangular windowed frames of 160 samples of 8kHz sampled speech (20ms frames). The autocorrelation method and Schur recursion produced a 12^{th} order synthesis filter whose parameters were transmitted as Log-Area Ratios quantised with 54 bits. The speech frame is then further divided into 4 subframes of 40 samples for the LTP and fixed codebook processing. The stochastic gain is quantised with 6 bits, which was not considered a limiting factor in the coder's performance.

The LTP could be switched into operation by setting a switch on the program command line. The LTP had an adaptive codebook of 147 elements giving 128 integer delays, and 128 non-integer delays. The precise allocation of delays and corresponding codebook index is listed in appendix 3. Delays could be modelled from 2.5 to 18.5ms. The LTP gain was quantised with 6 bits. Again this was not considered a limiting factor in overall coder performance. The precise bit allocation of all three speech coders is given in table 6.1.

Parameter	CELP1	CELP2	CELP3
12 LAR Coefficients	54	54	54
4 Codebook Indexes	32	32	36
4 Stochastic Gains	24	24	24
(4 LTP delays)	(32)	(32)	(32)
(4 LTP gains)	(24)	(24)	(24)
Total Bits per 20ms Frame	110	110	114
	(166)	(166)	(170)

Table 6.1 *Bit Allocation of Reference CELP coders CELP1, CELP2 and CELP3, Values in Parenthesis Correspond to Inclusion of the LTP.*

The objective performance of all three coders was measured with both gaussian and ternary codebooks. The performance of all six variants of coder is shown in table 6.2. Suffix "G" and "T" refer to gaussian or ternary codebooks respectively. The figures in parenthesis correspond to inclusion of the LTP. The gaussian tests were repeated four times with different sequences and the results averaged. The ternary results were taken once and the codebook sequence was a portion of the ternary sequence used in the US federal standard [12].

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)
CELP1G	5.99 (10.01)	3.67 (3.48)
CELP1T	6.21 (10.08)	3.45 (3.39)
CELP2G	6.21 (10.22)	3.52 (3.41)
CELP2T	6.48 (10.20)	3.37 (3.37)
CELP3G	6.64 (10.55)	3.51 (3.38)
CELP3T	6.83 (10.50)	3.46 (3.38)

Table 6.2 *Performance of Reference CELP Coders. Suffix "G" and "T" Refer to Gaussian or Ternary Codebooks Respectively. Values in Parenthesis Correspond to Inclusion of the LTP.*

Comparison of these results with those from chapter 6 highlight the dramatic improvement of analysis-by-synthesis methods. Without a LTP, the SNRSEG is increased from 1.33dB to 5.99dB and with a LTP it is increased from 4.03dB to 10.01dB. This significant improvement is echoed in informal listening tests. The inclusion of the perceptual error weighting further improves the perceived speech quality and this improvement increases with codebook size. This follows since with few available candidate vectors, it is more likely that the best non-error weighted vector is the same vector as best error weighted one, as the codebook size is increased it becomes more likely that the error weighted optimisation finds a better, different excitation vector than a non-error weighted procedure would.

The Log-Spectral Distance results are less conclusive. Without the LTP, they show a very slight improvement in quality (a drop from 3.77dB to 3.67dB) from the use of analysis-by-synthesis optimisation methods, which does not reflect the very significant improvement in subjective quality. With the LTP there is a significant increase in the spectral distance reading (from 2.95dB to 3.48dB) indicative of a worsening in quality which totally conflicts with the improvement in subjective quality. Clearly the log-spectral distance measure has proved unreliable in this case!

Study of the objective performance of the three coders without the LTP, shows a slight improvement in output SNRSEG (of 0.2dB) from coder CELP1G to CELP2G as the spacing of adjacent candidate vectors is increased from 1 to 2. There is also a slight improvement in output SNRSEG (of 0.4dB) from CELP2G to CELP3G as the number of candidate codebook vectors is increased from 256 to 512. These improvements are also present when the LTP is incorporated although of smaller magnitude. However, They are negligible in comparison with the improvement resulting from analysis-by-synthesis methods. With all three reference CELP coders, the ternary codebook has greater output SNRSEG of about 0.2 dB. With all the six variants of coder, both the SNRSEG and the log-spectral distance are significantly improved by inclusion of the LTP.

Informal listening tests show there is a very dramatic improvement arising from analysis-by-synthesis methods. The inclusion of an LTP gives excellent quality suitable for a practical scheme. Subjective improvements from CELP1G to CELP3G from the increase in codebook spacing and numbers of candidate vectors are just detectable with careful listening. The distortion with the gaussian codebook can be described as a gravelly sound, whereas the ternary codebook gives rise to a background crackle. The synthetic speech from the two codebooks is obviously different but it is difficult to detect which is superior.

Synthetic speech without inclusion of the LTP has a much inferior quality far below that required for a practical scheme. Without LTP the gaussian coders CELP1G to CELP3G sound very gravelly with no detectable difference between their outputs. The distortion arising from the ternary codebooks without LTP is a heavy growl, again there was no detectable difference between output from all three ternary coders.

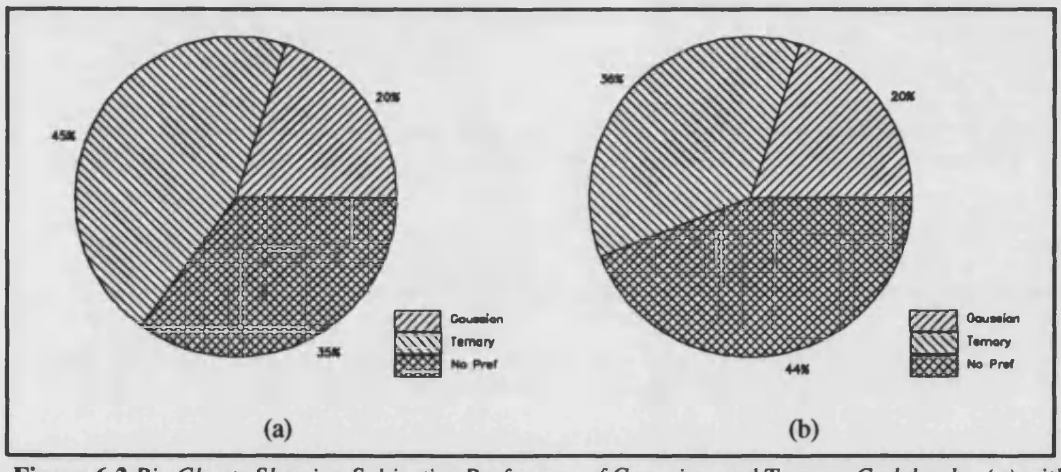

Figure 6.2 *Pie Charts Showing Subjective Preference of Gaussian and Ternary Codebooks, (a) with LTP, (b) without LTP.*

Figure 6.2 shows the results of paired comparison listening tests comparing the subjective preference of gaussian and ternary codebooks. These results were taken for coder CELP3, both with and without the LTP. In both cases, there is a clear preference for synthetic speech derived from the ternary codebook, (with LTP, 45% versus 20% and without LTP, 36% versus 20%).

For reference, performance of all six variants of coder was assessed in noisy channels. The transmitted bits were corrupted randomly with varying error rates and the resulting SNRSEG and log-spectral distance are graphed in the figures 6.3 to 6.6. Figures 6.3 and 6.4 show the gaussian CELP coder without and with the LTP respectively and figures 6.5 and 6.6 show the ternary CELP coder without and with the LTP respectively. The major difference over these figures is the rate of degradation with and without inclusion of the LTP. On the SNRSEG graphs, curves cross the *BER* axis at about 0.8%

with the LTP whereas without the LTP, curves cross the *BER* axis at about 2.0% BER. This indicates the LTP parameters are very error sensitive. There is a slight variation in the crossing of the *SNRSEG* axis due to the differences in coder clear channel performance. The only other difference is the variation of point cluster size between gaussian curves and ternary curves. The variation in cluster size cannot be attributed to the speech coder type, since points for gaussian curves correspond to different initialisation sequences, whereas those for ternary curves correspond to the same initialisation sequence with readings repeated four times.

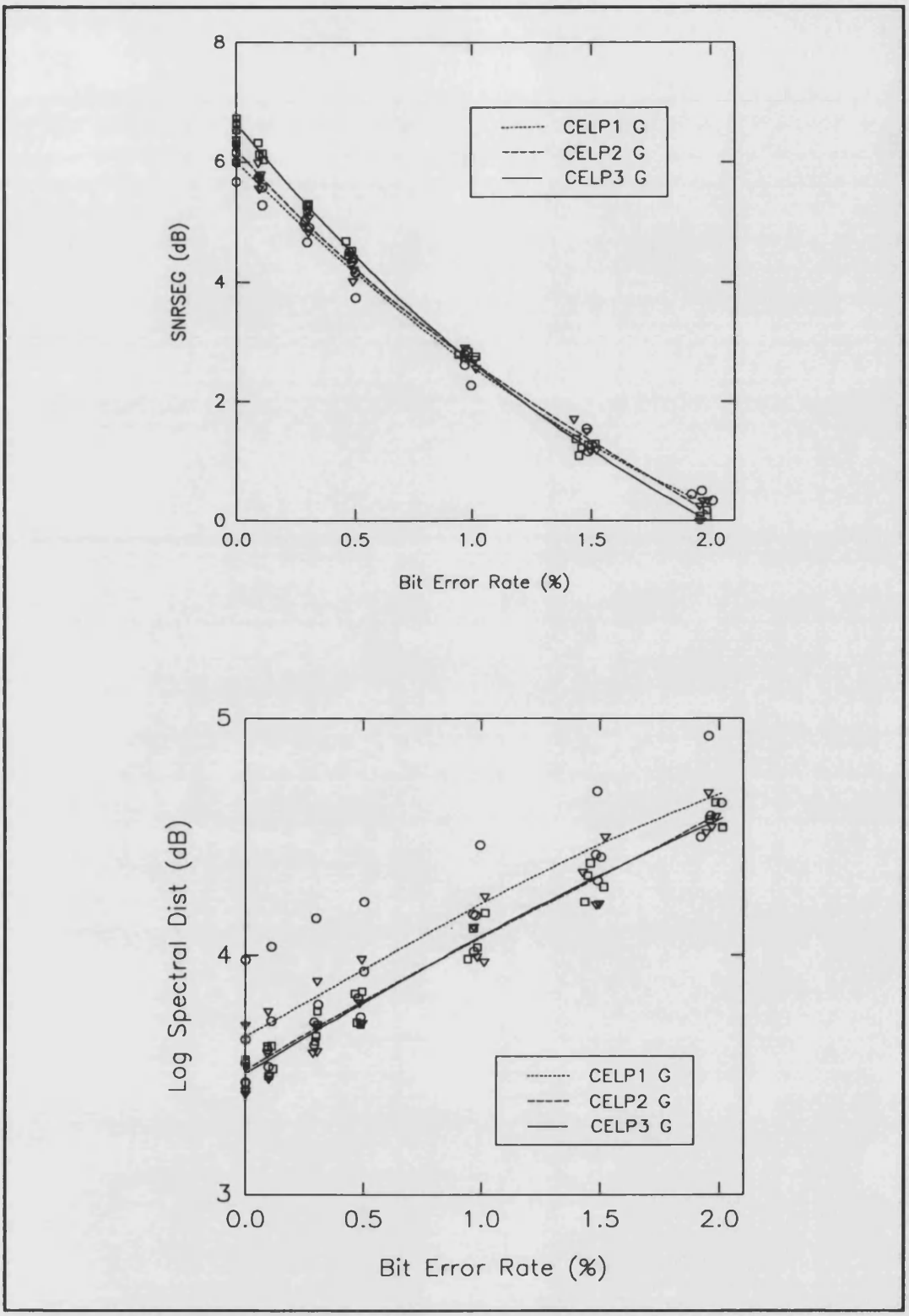

Figure 6.3 *Performance of Reference CELP Coders with Gaussian Codebook without LTP in Channel Errors.*

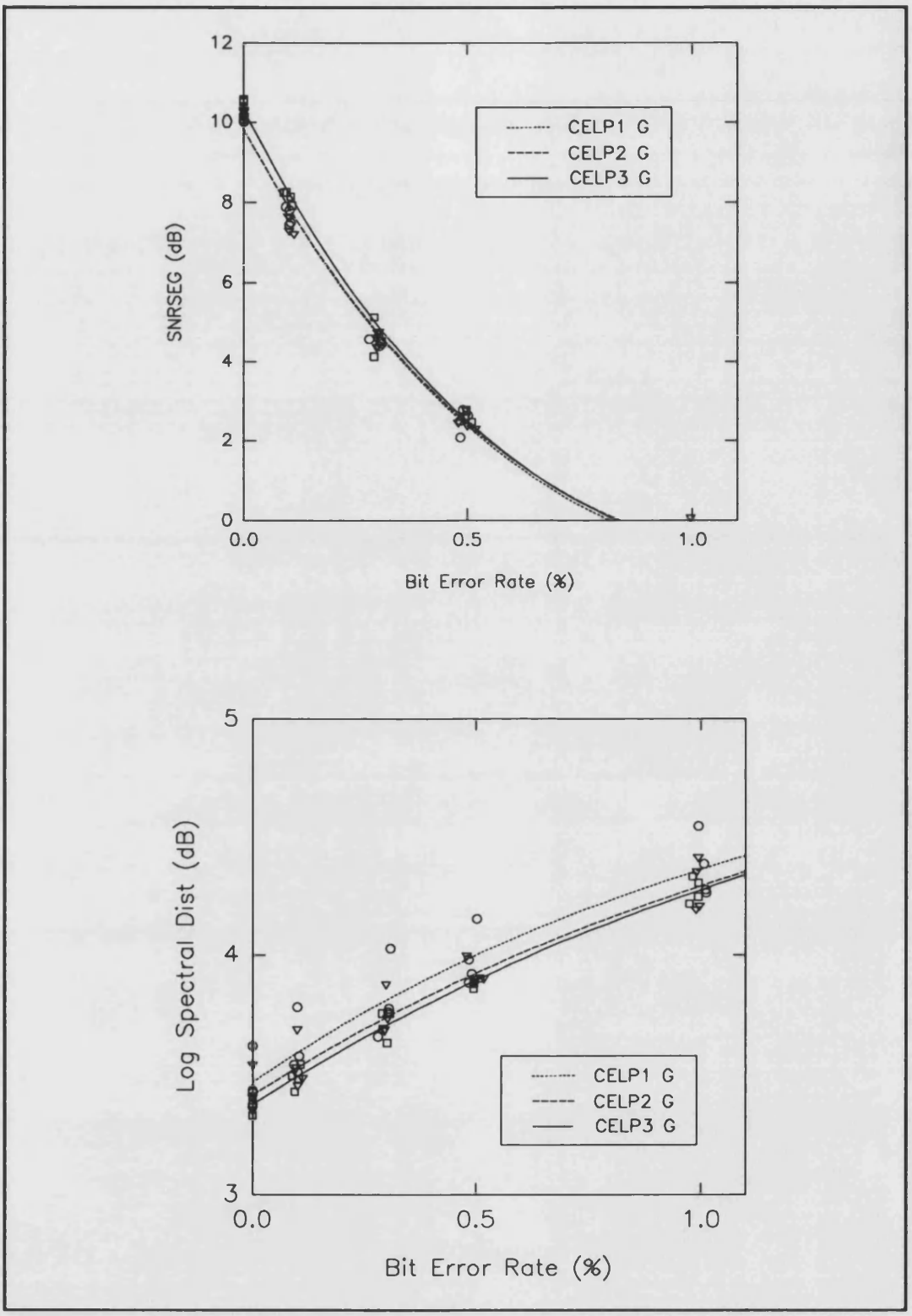

Figure 6.4 *Performance of Reference CELP Coders with Gaussian Codebook with LTP in Channel Errors.*

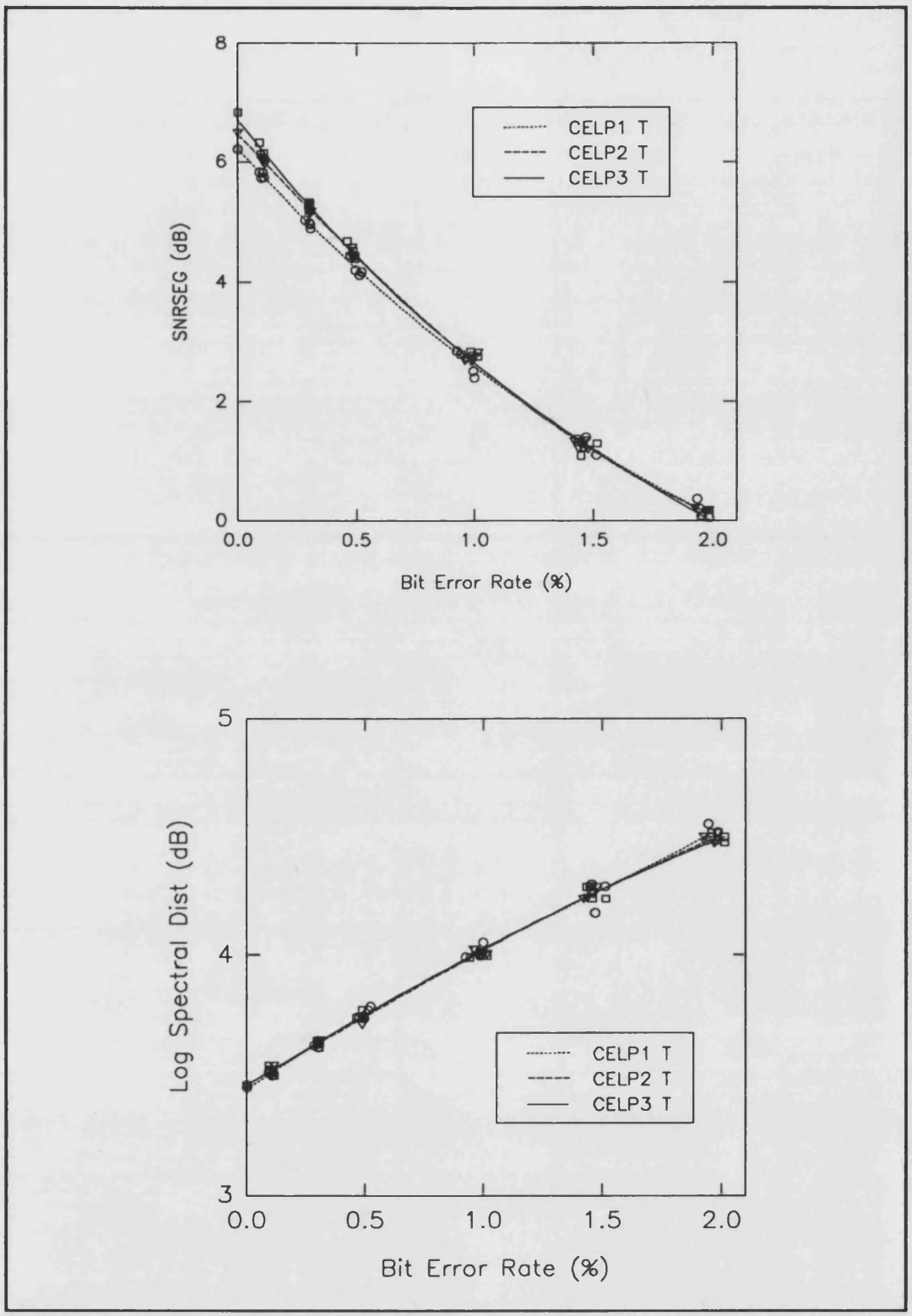

Figure 6.5 *Performance of Reference CELP Coders with Ternary Codebook without LTP in Channel Errors.*

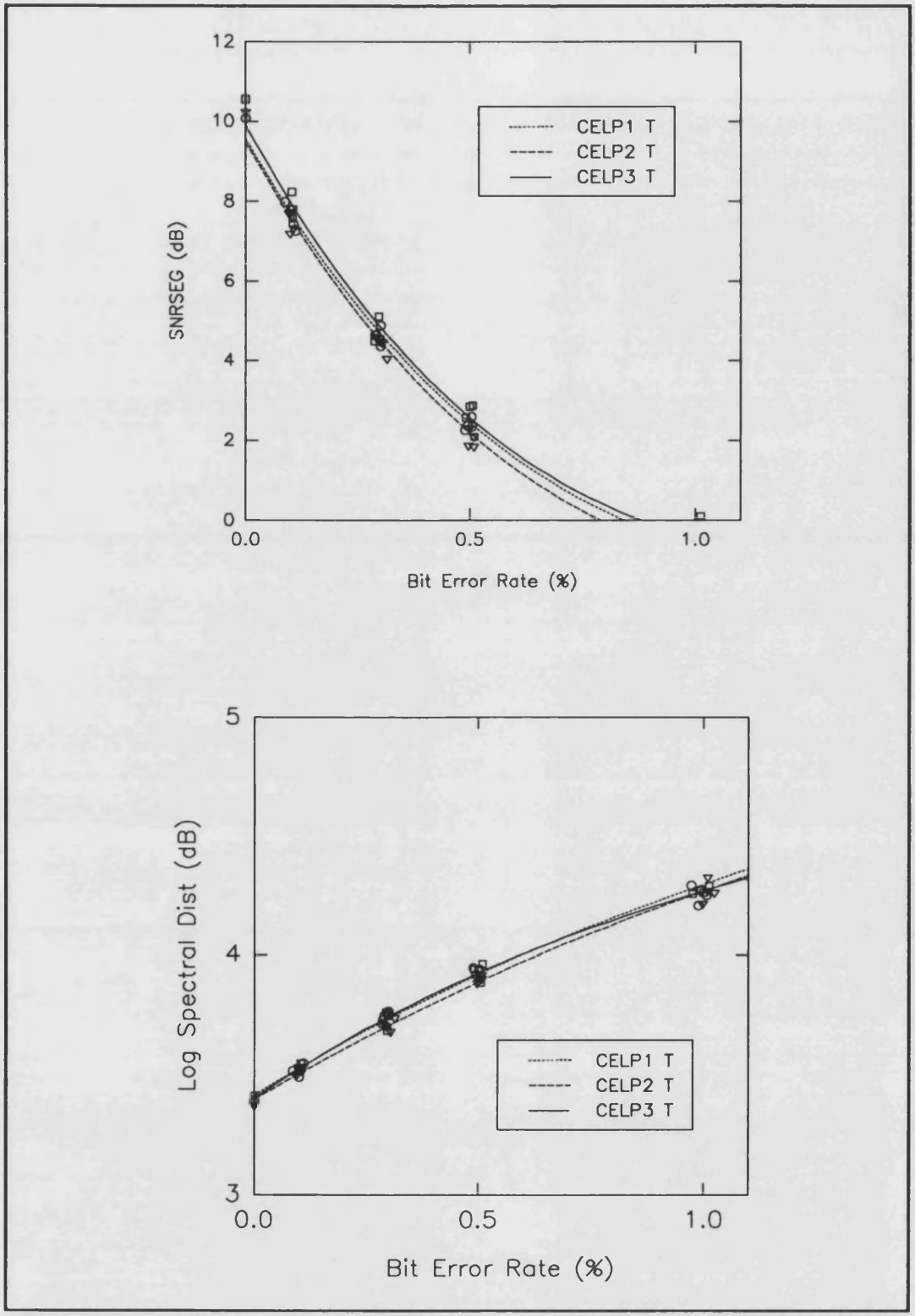

Figure 6.6 *Performance of Reference CELP Coders with Ternary Codebook with LTP in Channel Errors.*

6.3 Analysis-by-Synthesis Pure SEV

This section develops an analysis-by-synthesis version of the Pure SEV from chapter 5. Again it is based on the speech coder module of appendix A1.2, sharing much of the same code as the reference CELP coders just described.

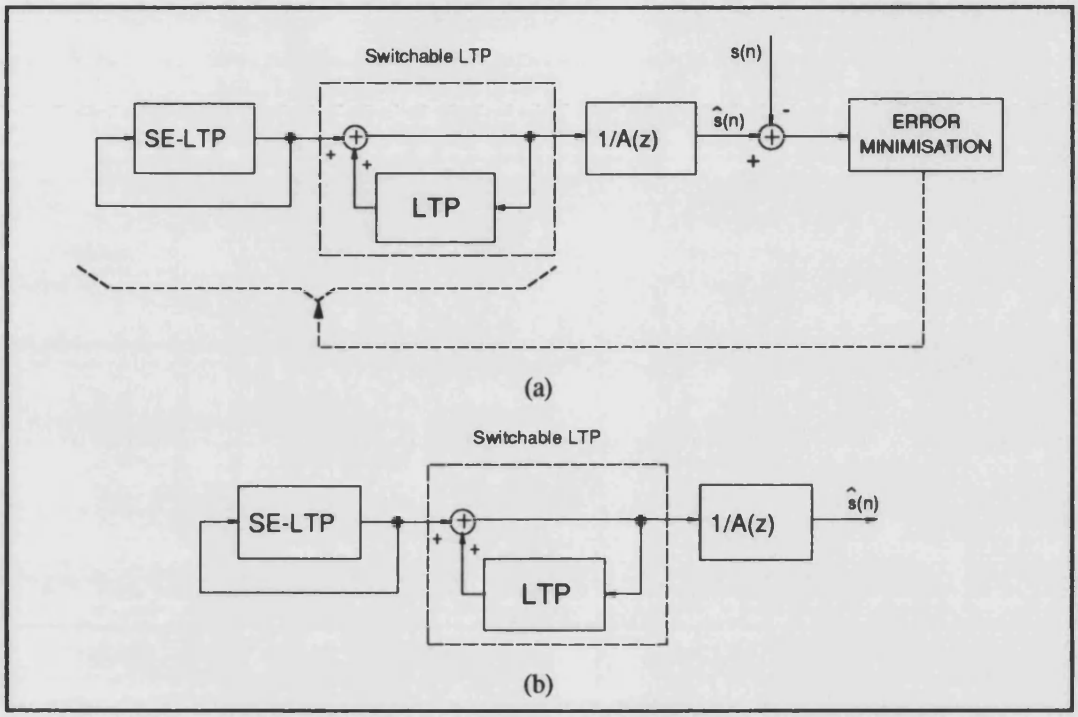

Figure 6.7 *Analysis-by-Synthesis Pure SEV with Switchable LTP, (a) encoder, (b) decoder.*

The analysis-by-synthesis SEV is depicted in figure 6.7, (a) the encoder and (b) the decoder. It consists of a self-exciting LTP (SE-LTP) which, after initialisation, maintains an adaptive codebook updated solely from its own output. The chosen adaptive codebook excitation vector is amplified/attenuated before input to the next stage. If enabled, this is an LTP which models the pitch period and its output excites the short-term synthesis filter $1/A(z)$ giving output speech. Otherwise, without the LTP, the SE-LTP output is used to excite the synthesis filter $1/A(z)$ directly. The block labelled "error minimisation" refers to the selection of optimum SE-LTP and LTP parameters. In practice the LTP parameters are optimised first. When the perceptual error weighting is enabled, this error minimisation occurs after the difference signal

has been passed through the conventional linear prediction perceptual error weighting filter $A(z)/A(bz)$, where $b = 0.8$ [2]. The SEV decoder is depicted in figure 6.7(b). This is entirely a sub-component of the encoder, as is the case with analysis-by-synthesis speech coders. The optimum parameters are transmitted to the SE-LTP and LTP sections which re-create the residual, this exciting the synthesis filter $1/A(z)$ giving synthetic speech.

The synthetic speech output from the Pure SEV, incorporating the LTP, is given by

$$\hat{s}(n) = \gamma v_l(n - d_l) * h(n) + \zeta v_s(n - d_s) * h(n) + \hat{m}(n) \qquad 6.3$$

where v_l, d_l and γ are the LTP adaptive codebook buffer, delay and gain respectively and v_s, d_s and ζ are the SE-LTP codebook buffer, index and gain respectively. $h(n)$ is the impulse response of the synthesis filter $1/A(z)$ and $\hat{m}(n)$ is the contribution due to its memory. Minimisation of the mean squared error gives

$$E = \sum_{n=0}^{N-1} \tilde{s}^2 - \gamma \sum_{n=0}^{N-1} \tilde{s}(n)\,[v_l(n - d_l) * h(n)] - \zeta \sum_{n=0}^{N-1} \tilde{s}(n)\,[v_s(n - d_s) * h(n)] \qquad 6.4$$

where N is the analysis subframe length. When the LTP is not incorporated, the contribution due to the LTP is ignored in the above equations and the SE-LTP stage parameters ζ and d_s optimised accordingly. Alternatively, when the LTP is incorporated, joint optimisation of the mean squared error for all possible combinations of d_s and d_l would present an excessive computational task. Instead they are optimised sequentially, with LTP parameters γ and d_l optimised first, followed by fixed codebook parameters ζ and d_s.

Linear Predictive analysis is performed on non-preemphasised, non-overlapping rectangular windowed frames of 160 samples of 8kHz sampled speech (20ms frames). The autocorrelation method and Schur recursion produced a 12th order synthesis filter

whose parameters were transmitted as Log-Area Ratios quantised with 54 bits. The speech frame is then further divided into 4 subframes of 40 samples for LTP and SE-LTP processing.

The inclusion of a conventional LTP was enabled by setting a switch on the program command line. The LTP was identical to that used in the reference CELP coders. It had an adaptive codebook of 147 elements giving 128 integer delays, and 128 non-integer delays. The precise allocation of delays and corresponding codebook index is listed in appendix 3, this allowed modelling of delays from 2.5 to 18.5ms. The gain term of this LTP was quantised with 6 bits. Again this was not considered a limiting factor in overall coder performance.

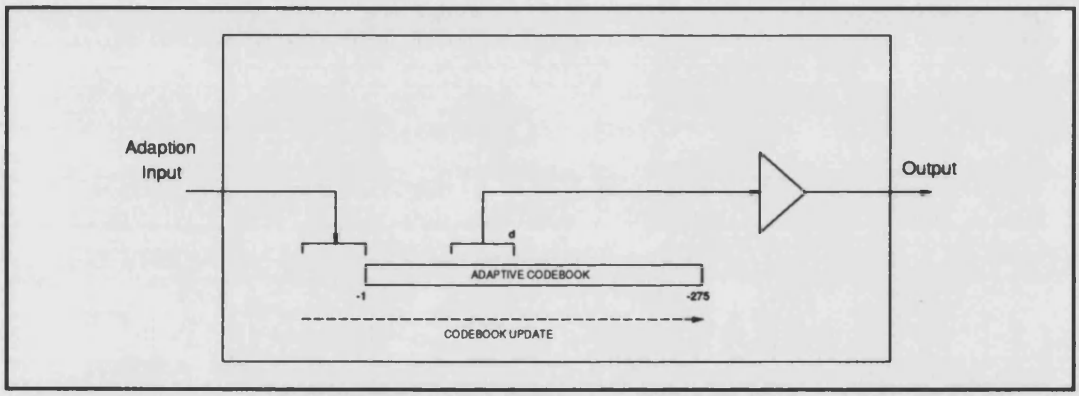

Figure 6.8 *Pure SEV Self-Exciting LTP (SE-LTP).*

Figure 6.8 shows a more detailed diagram of the SE-LTP stage. It consists of a 275 element adaptive codebook,

$$\{v_{s-i}\} = \{v_{s-275}, v_{s-274}, v_{s-273}, \ldots, v_{s-1}\} \qquad 6.5$$

with a sliding window capable of extracting 256 candidate excitation vectors of length 40 samples with delays d_s from 20 to 275 samples, corresponding to a minimum pitch period of 2.5ms (400Hz). The adaptive codebook was initialised with a zero-mean, gaussian distributed random sequence at the start of a coding session.

From this adaptive codebook, Delays d_s of 40 to 275 are available directly as adjacently spaced codebook entries corresponding to elements $[v_{s-40}..v_{s-1}]$, $[v_{s-41}..v_{s-2}]$ to $[v_{s-275}..v_{s-236}]$ respectively. Whereas delays d_s of 20 to 39 are obtained by cyclically repeating codebook elements from $[v_{s-d_s}..v_{s-1}]$ to form a 40 element vector. For example, delay 20 corresponds to $[v_{s-20}..v_{s-1}, v_{s-20}..v_{s-1}]$, delay 21 corresponds to $[v_{s-21}..v_{s-1}, v_{s-21}..v_{s-3}]$ and delay 39 corresponds to $[v_{s-39}..v_{s-1}, v_{s-39}]$. Once an excitation vector has been extracted, it is amplified by the SE-LTP gain and then used both as input to the next coder stage and for codebook adaption. For convenience, this vector is placed in the notional continuation of the adaptive codebook $[v_{s0}..v_{s39}]$ and in the Pure SEV, when the SE-LTP delay is greater than or equal to 40, is given by

$$v_s(i) = \zeta v_s(i - d_s) \qquad 0 \le i \le 39, d_s \ge 40 \qquad 6.6$$

where ζ is the SE-LTP gain, calculated from equation 3.72 and d_s is the SE-LTP delay. Alternatively when d_s is in the range 20 to 39, the output is given by the pair of equations

$$v_s(i) = \zeta v_s(i - d_s) \qquad \begin{matrix} 0 \le i < d_s \\ 20 \le d_s \le 39 \end{matrix} \qquad 6.7a$$

$$v_s(i) = v_s(i - d_s) \qquad \begin{matrix} d_s \le i \le 39 \\ 20 \le d_s \le 39 \end{matrix} \qquad 6.7b$$

The next process is the left shifting of the entire codebook buffer by 40 samples, shifting the adaption vector into the codebook

$$v_s(i) = v_s(40 + i) \qquad -275 \le i < 0 \qquad 6.8$$

The SE-LTP gain is encoded using 6 bits, this was not considered a limiting factor in overall coder performance. The precise bit allocation of this Pure SEV is given in table 6.3.

Parameter	Number of Bits
12 LAR Coefficients	54
4 SE-LTP delays	32
4 SE-LTP gains	24
(4 LTP delays)	(32)
(4 LTP gains)	(24)
Total Bits per 20ms Frame	110 (166)

Table 6.3 *Bit Allocation of Pure SEV, Values in Parenthesis Correspond to Inclusion of the LTP.*

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)
Pure SEV	6.83 (9.96)	3.60 (3.45)
CELP1G	5.99 (10.01)	3.67 (3.48)

Table 6.4 *Performance of the Analysis-by-Synthesis Pure SEV, Values in Parentheses Correspond to the Inclusion of the LTP.*

The objective performance of this coder was measured and the results are shown in table 6.4. Compared to the low complexity SEVs of chapter 5: Without LTP, there is a threefold improvement in the SNRSEG (from 2.26dB to 6.83dB), and with LTP, a two and a half times improvement (from 4.03db to 9.96dB). Comparing the output SNRSEG for this Pure SEV and the reference coder CELP1G: Without LTP, the SEV is significantly higher than the CELP coder (6.83dB versus 5.99dB), with LTP the two coders perform equally well (9.96dB versus 10.01dB). The output waveform of this coder with LTP over the short utterance "Seven" is shown in figure 6.9, showing excellent reproduction of the utterance pitch.

Without LTP, informal listening tests showed the SEV speech quality was notably better than the reference CELP coder. With the LTP, the subjective preference of reference coder CELP1G and the Pure SEV was tested by paired comparison and the

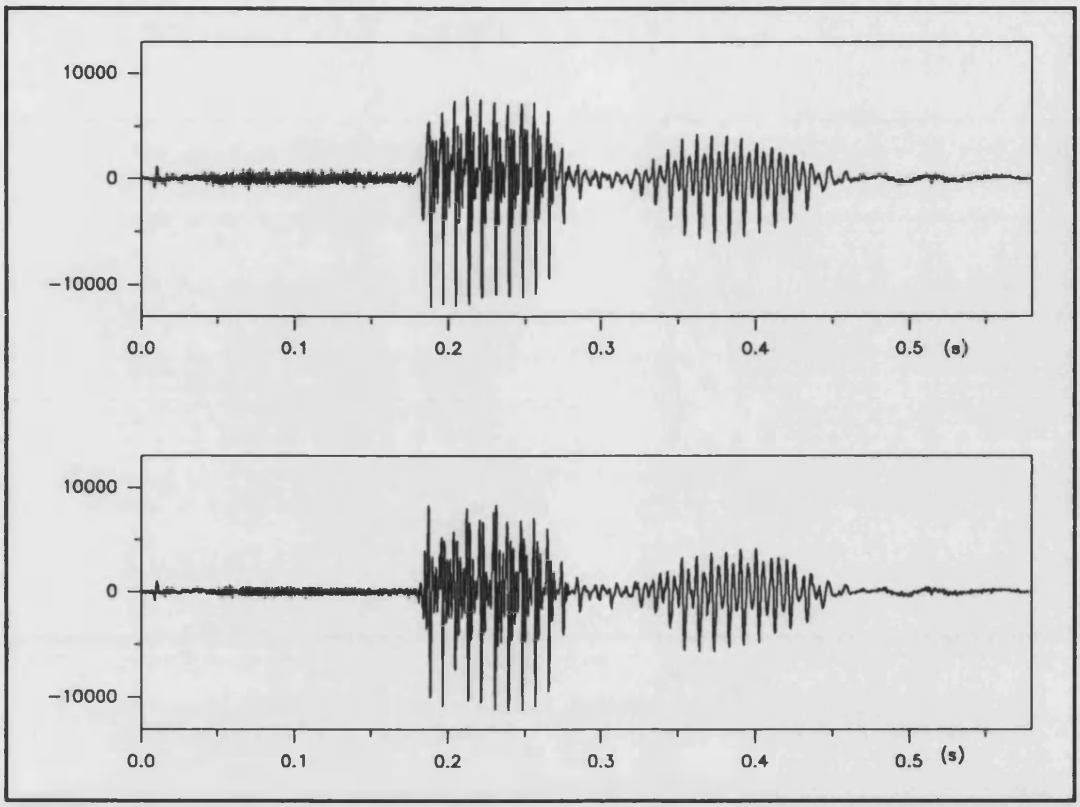

Figure 6.9 *Performance of this Analysis-by-Synthesis Pure SEV, the Input shown in the Top Trace is the Utterance "Seven" by a Male Speaker, Bottom Trace is the Coder Output.*

result shown in figure 6.10. This result is disappointing, contrary to the results of the previous chapter, the CELP coder has a clearly higher subjective quality than the Pure SEV (73% versus 16%). This was surprising and contradicted earlier informal listening!

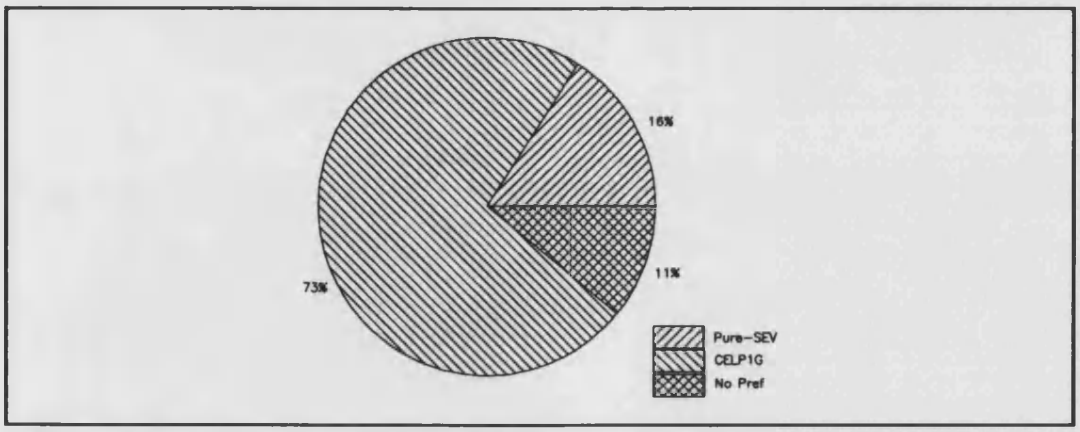

Figure 6.10 *Pie Chart Showing Subjective Preference of Pure SEV and Reference CELP Coder.*

The Log-Spectral distance results are inconclusive. Comparing this vocoder to the low complexity (non analysis-by-synthesis) version of the previous chapter: without LTP, figures are approximately equal and with LTP, are worse (3.01dB versus 3.45dB). This worsening in quality contradicts both the perceived subjective quality and the SNRSEG measure. The coder has less effectively modelled the speech spectrum but has resulted in a lower noise output. The lower noise output is subjectively preferable.

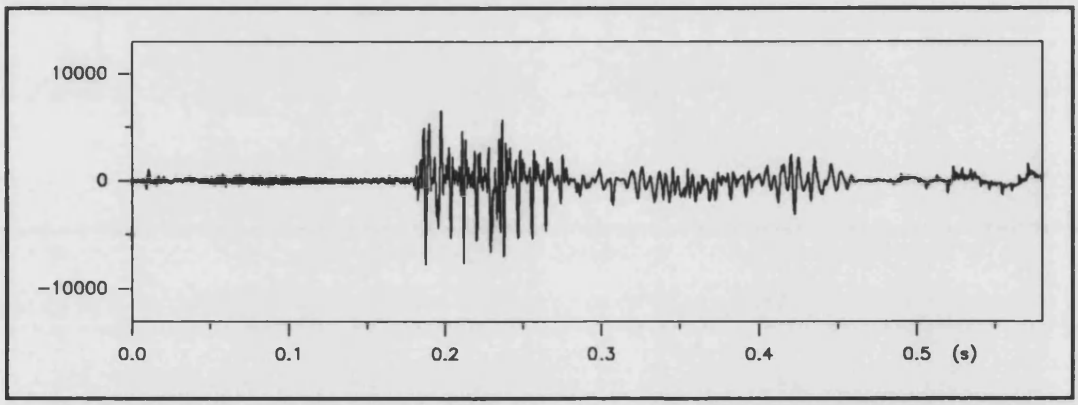

Figure 6.11 *Performance of this Pure SEV with the Same Utterance "Seven" in the Presence of Channel Errors, Only Excitation Parameters are Corrupted with BER of 0.4%.*

Although some promising results are obtained in error free transmission, the usefulness of this analysis-by-synthesis Pure SEV is limited as the waveform of figure 6.11 illustrates (this is the same utterance "Seven" used in figure 6.9). It shows the effect of corrupting the excitation parameters (SE-LTP and LTP gains and delays) with a BER of 0.4%. Over this very short utterance, the serious degradation of output is clearly visible as it propagates throughout the utterance, this is well established after only 0.5s. This results from loss of identity between adaptive codebooks of speech encoder and decoder. Once lost, this identity can never be regained. For this reason the pure SEV alone can never form the basis of a practical scheme. The next section introduces two methods of rectifying this disastrous error performance.

6.4 Error Robust Initialisation Schemes

The disastrous error performance of the Pure SEV identified in the previous section must be overcome for a practical speech coding scheme. Should identity be lost between adaptive codebooks of encoder and decoder, it must be quickly restored. A mechanism is required to regularly force the decoder into a known state. A possible method would be to reserve one bit of each subframe as an initialisation bit. When set, the contents of the decoder adaptive codebook would be reset to a known startup sequence ending any error propagation. If the overall bit rate of the SEV is to remain unaltered, the number of excitation vectors available from the adaptive codebook must be halved.

The SEV could then be initialised every given number of frames. When initialised it functions as a CELP coder and as demonstrated by the previous section lower performance can be expected. However there are likely to be subframes where one of the fixed initialisation sequences performs better, such as after a pause or in the gaps between words. For this reason, a better approach is to only initialise the SEV when beneficial to the synthetic speech quality. Therefore the search for the optimum excitation vector must include both the candidate adaptive vectors and the fixed vectors. Taking this approach further, rather than having a specific initialisation bit, the codebook is composed of both adaptive and fixed candidate excitation sequences. Two possible methods are studied in this section, these being the Partial Fixed Codebook (PFC) SEV and the Separate Fixed Codebook (SFC) SEV.

6.4.1 Partial Fixed Codebook (PFC) SEV

The Partial Fixed Codebook (PFC) SE-LTP, depicted in figure 6.12, modifies the Pure SEV, restricting the codebook adaption process over less of the codebook buffer. Thus the range of operation of equation 6.8 is modified becoming

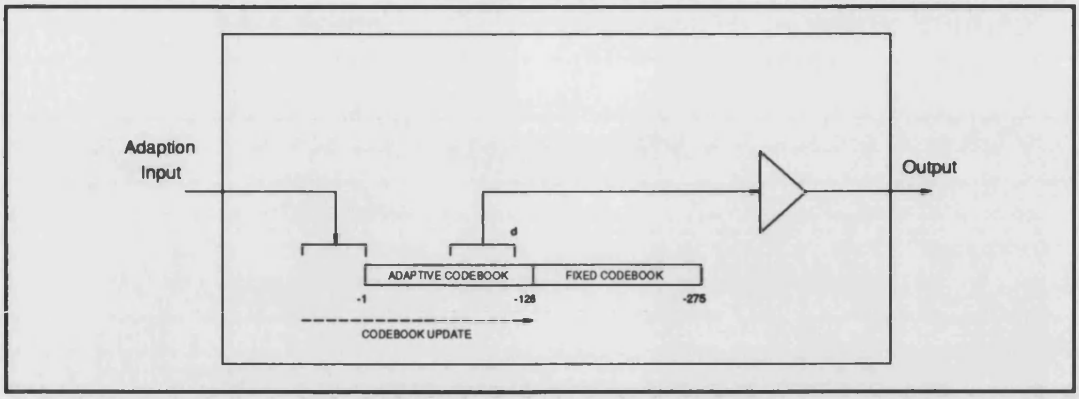

Figure 6.12 *Partial Fixed Codebook SE-LTP.*

$$v_s(i) = v_s(40+i) \qquad -128 \le i < 0 \qquad 6.9$$

The codebook is divided into part fixed and part adaptive. Its length remained 275 samples, the lowest 147 values $[v_{s-275}..v_{s-129}]$ are loaded with the random initialisation sequence at the start of the coding session, giving 108 fixed initialisation vectors that are unaltered throughout coder operation. An adaptive codebook of 128 samples remains $[v_{s-128}..v_{s-1}]$, this has 109 possible delays from 20 to 128 and models pitch periods of 2.5-13.5ms. There remain 39 possible delays at the transition where vectors are part fixed and part adaptive. The entire codebook is searched as before for the optimum excitation vector and both fixed and adaptive sequences are considered. An initialisation vector is chosen if it gives a superior representation of the original speech than all the adaptive vectors. Proportionate lengths of both fixed and adaptive codebook could be altered, but this has not been investigated. This is the technique used by Menez *et al* in their recently proposed ACELP schemes [33][34].

6.4.2 Separate Fixed Codebook (SFC) SEV

The Separate Fixed Codebook (SFC) SE-LTP separates the predictor codebook into two distinct codebooks, one is purely for initialisation and the second is a normal adaptive codebook. For programming convenience, both codebooks used the same buffer. The codebook buffer length was increased to 314 samples, the lowest 167 values $[v_{s-314}..v_{s-148}]$ are loaded with the random initialisation sequence at the start of the

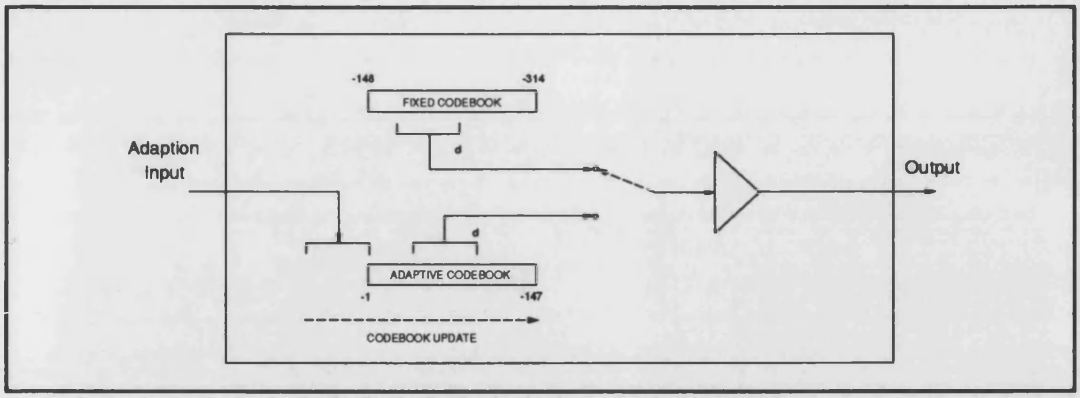

Figure 6.13 *Separate Fixed Codebook SE-LTP.*

coding session, giving 128 initialisation vectors. The remainder of the codebook forms the adaptive codebook $[v_{s-147}..v_{s-1}]$ giving 128 possible delays from 20 to 147 samples. The codebook search procedure now works in two stages, delays 20 to 147 (adaptive) and delays 187 to 314 (initialisation). Delays corresponding to initialisation vectors are transmitted as delay-59 and delays corresponding to adaptive vectors are transmitted as delay-20 to keep the delay parameter transmitted as an 8 bit value. Equation 6.8 becomes

$$v_s(i) = v_s(40+i) \qquad -167 \le i < 0 \qquad 6.10$$

This initialisation technique has been used by Rose *et al* [43] in their SEV work.

6.4.3 Performance of Initialisation Schemes

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)	Initialisation Rate (%)
Pure SEV	6.83 (9.96)	3.60 (3.45)	-
PFC-SEV	6.67 (10.01)	3.71 (3.54)	28.2 (39.5)
SFC-SEV	6.60 (10.01)	3.74 (3.49)	32.9 (48.7)
CELP1G	5.99 (10.01)	3.67 (3.48)	-

Table 6.5 *Objective Performance and Initialisation Rate of Pure SEV, PFC-SEV, and SFC-SEV, Values in Parentheses Correspond to Inclusion of the LTP.*

Objective performance of both initialisation schemes was measured over the usual test record of 20 speakers. Each test was repeated four times, each time with a different initialisation sequence (all gaussian). The average results are shown in table 6.5, along with those for the Pure SEV and reference CELP1G. Without LTP: Both initialisation methods give a small loss in output SNRSEG, of the order of 3% less than the Pure SEV. However, output SNRSEG remains 10% greater than that of reference coder CELP1G. The initialisation rate is lower for the PFC-SEV. This follows since any partial initialisation due to the transition in the codebook, does not force the decoder into an entirely known state and is not counted.

With LTP: Both initialisation methods perform equally well at 10.01db, which is identical to the reference coder CELP1G. There is no longer any superiority for the SEV. Also the initialisation rate is significantly higher with and approaches 50% for the SFC-SEV. Since the fixed and adaptive portions of the SE-LTP codebook are of equal length, equal numbers of excitation vectors are coming from both fixed and adaptive portions, indicating that the SE-LTP now plays little part in the modelling of speaker pitch.

Objective performance of these initialisation methods in noisy channels, as the BER varies, is shown figures 6.14 and 6.15, where both SNRSEG and log-spectral distance are plotted. Figure 6.14 shows performance of both schemes without LTP and figure 6.15 with LTP. Also plotted on each graph is the performance of reference coder CELP1G. The graphs show that, unlike the Pure SEV, both schemes are capable of tolerating some channel errors and there is little performance difference between them. Both with and without LTP, the reference CELP coder maintains a higher output SNRSEG as the BER is increased than both initialisation methods. Without LTP, the SEVs SNRSEG advantage is maintained up to 0.2% BER.

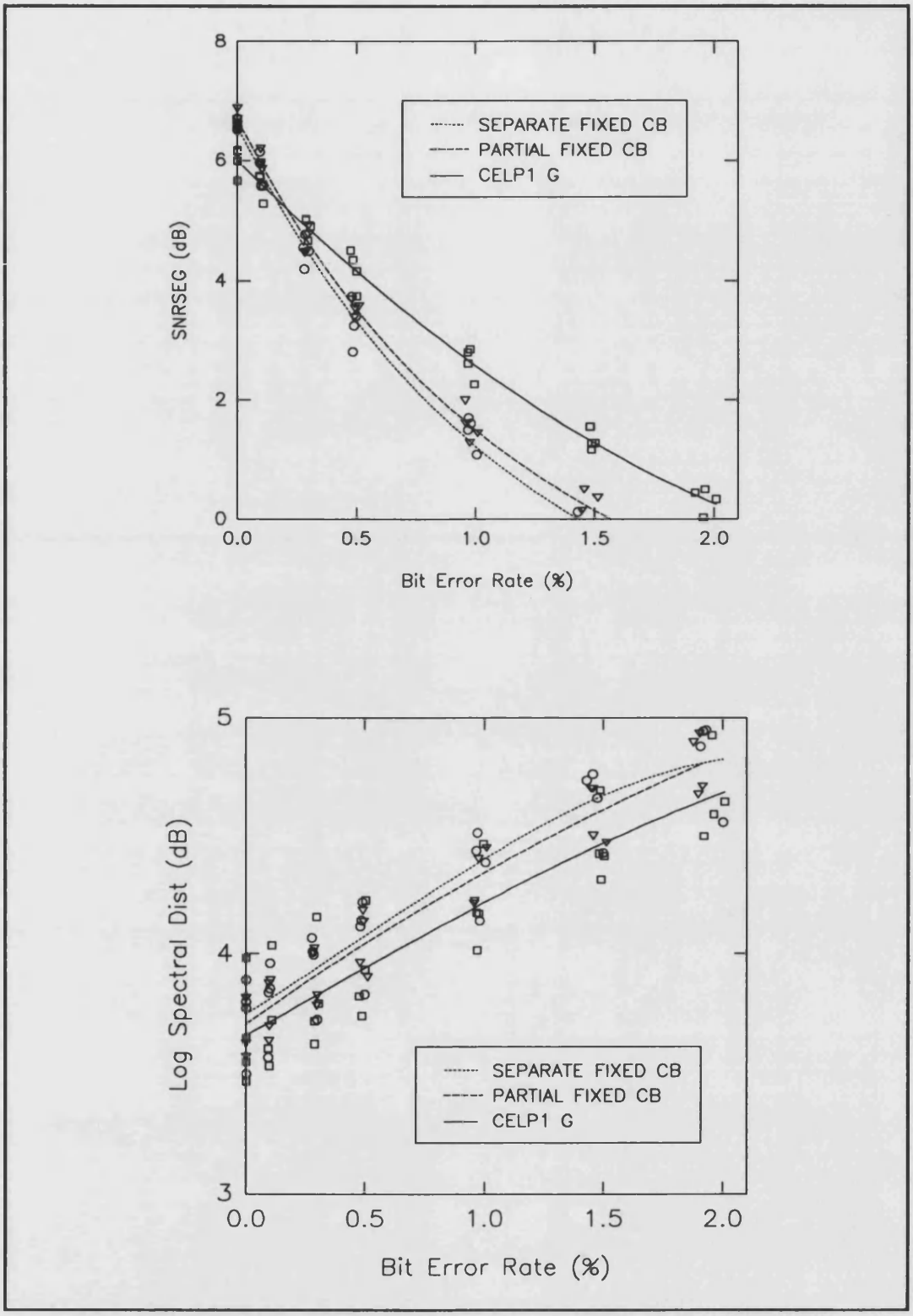

Figure 6.14 *Performance of PFC-SEV and SFC-SEV in Channel Errors, without LTP.*

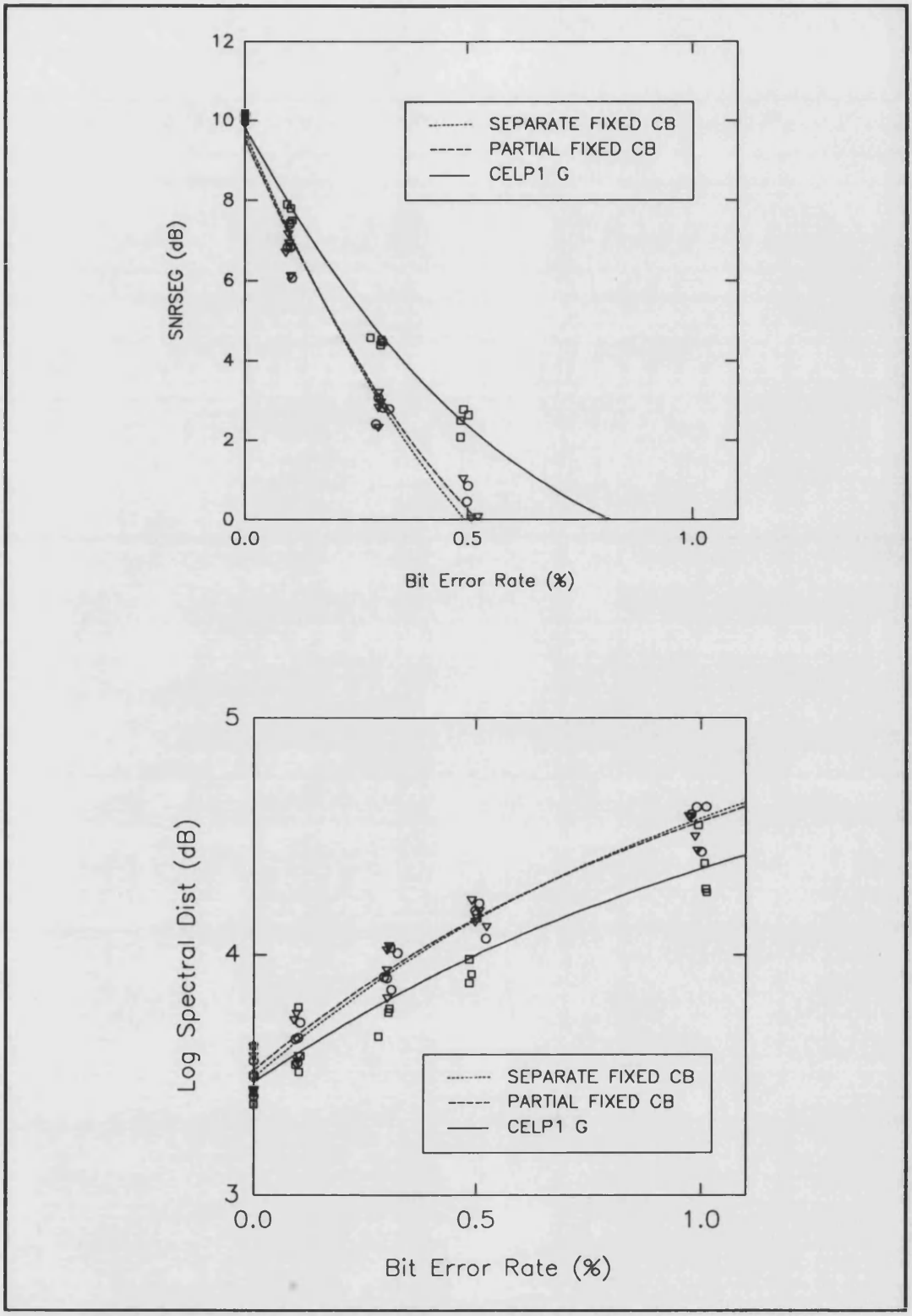

Figure 6.15 *Performance of PFC-SEV and SFC-SEV in Channel Errors, with LTP.*

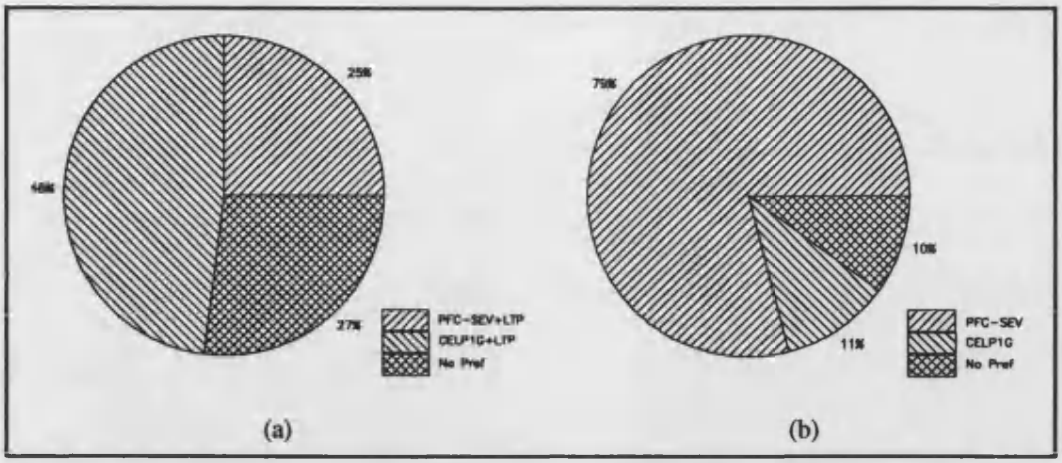

Figure 6.16 *Pie Charts Showing Subjective Preference of PFC-SEV and CELP1G, (a) with LTP, (b) without LTP.*

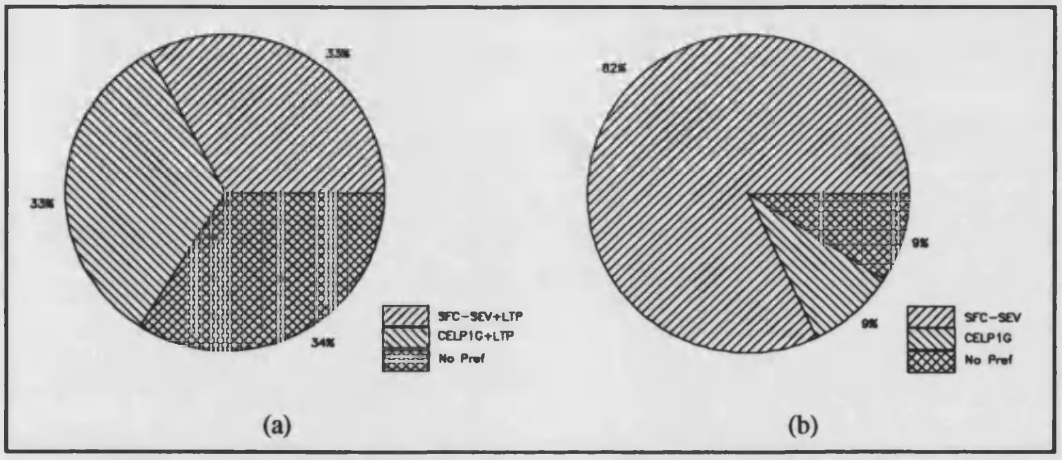

Figure 6.17 *Pie Charts Showing Subjective Preference of SFC-SEV and CELP1G, (a) with LTP, (b) without LTP.*

Paired comparison subjective testing was used to compare output produced by the SEVs with that of the reference coder CELP1G. In figure 6.16, reference coder CELP1G and the PFC-SEV are compared and in figure 6.17, reference coder CELP1G and the SFC-SEV are compared. Each test was performed by 50 listeners hearing two pairs of sentences, one pair from a male speaker and one pair from a female speaker. Also each comparison was performed with and without the LTP. Without the LTP, synthetic speech from the PFC-SEV is unanimously preferred to that from reference CELP1G (79% versus 11%). However the preference is reversed with the LTP with the reference being clearly preferred to the SEV (48% versus 25%). Results from the SFC-SEV are

similar, without the LTP there is also a unanimous preference for the SFC-SEV compared to the reference (82% versus 9%). However with the LTP, there is no preference between synthetic speech outputs (33% versus 33% with 34% indicating no preference). These results agree with objective measures, showing a clear advantage of a SE-LTP stage over a fixed codebook stage, provided a LTP stage is not used.

6.5 Alternative Codebook Adaption

The previous sections have demonstrated that error robust performance of the SEV is possible. They also demonstrated the superior performance of a SE-LTP stage over a fixed codebook stage when a LTP is not used. However the performance in noisy channels at higher BERs is still significantly worse than that of the reference CELP coder. This section introduces two new adaption strategies for the codebook and studies their effect on coder error robustness.

The primary reason for the serious degradation in channel errors is the loss of identity between adaptive codebooks of speech encoder and decoder. The use of either the PFC of SFC in the SEV has allowed some tolerance to channel errors, but more is still required. Loss of identity arises from corruption to transmitted SE-LTP gain and delay parameters. In the SEV implementation of this thesis, without the LTP stage, these two parameters occupy over half the transmitted bits.

Two new adaptions have been developed which attempt to constrain the variation in adaptive codebook amplitude, by making adaption independent of transmitted SE-LTP gain, these are termed adaptions A and B. They give a number of advantages:

1. The transmitted gain is eliminated from the codebook adaption process, significantly improving error robustness.
2. Since codebook rms amplitude remains more uniform, a transmission error in a delay term results in less of a power difference between correct and incorrect excitation

vectors and hence less degradation to the synthetic speech.

3. A fixed number of gain quantisation levels covers a smaller range of gain values. Thus any quantised gain value is likely to be closer to the actual gain value.

Conventional codebook adaption follows equations 6.6 and 6.8/6.9/6.10, for the Pure SEV, PFC-SEV and SFC-SEV coders respectively and was discussed in detail in section 6.3. The new adaptions result in modification to equation 6.6.

$$v_s(i) = \zeta v_s(i - d_s) \qquad 0 \le i \le 39, d_s \ge 40 \qquad 6.6$$

The first method, Adaption A, simply removes the gain term, hence the adaption equation becomes

$$v_s(n) = v_s(n - d_s) \qquad 0 \le n \le 39, d_s \ge 40 \qquad 6.11$$

The second method, Adaption B, normalises the rms codebook amplitude equal to the rms amplitude of the fixed codebook. This is calculated at the start of the coding session. Assuming the SFC-SEV of section 6.4 is being used.

$$A = \sqrt{\sum_{k=-314}^{-148} v_s^2(k)} \qquad 6.12$$

The limits of the summation can be suitably modified for the PFC-SEV and Pure SEV cases. The constant A is then used in the adaption equation for normalisation of the codebook amplitude.

$$v_s(i) = v_s(i - d_s) \cdot \frac{A}{\sqrt{\sum_{k=0}^{39} v_s^2(k - d_s)}} \qquad 0 \le i \le 39, d_s \ge 40 \qquad 6.13$$

The PFC, SFC and Pure SEVs were modified such that the SE-LTP could be switched between any one of these three adaption schemes. The switchable adaption SFC SE-LTP

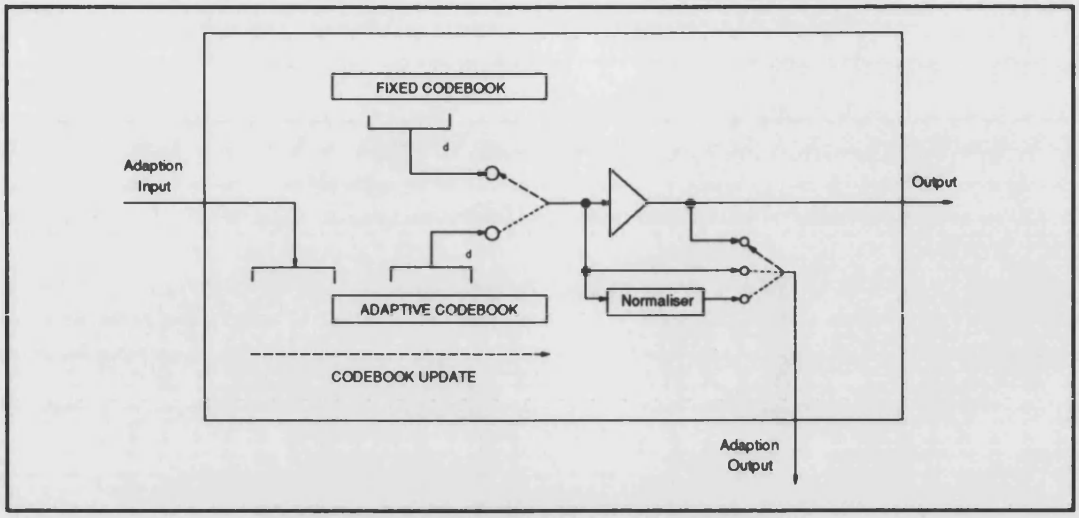

Figure 6.18 *SFC SE-LTP with Switchable Adaption.*

is depicted in figure 6.18, showing a switch selecting one of three sources for the adaption output. Using this modified SE-LTP stage, the SEV encoder block diagram is modified to that of figure 6.19.

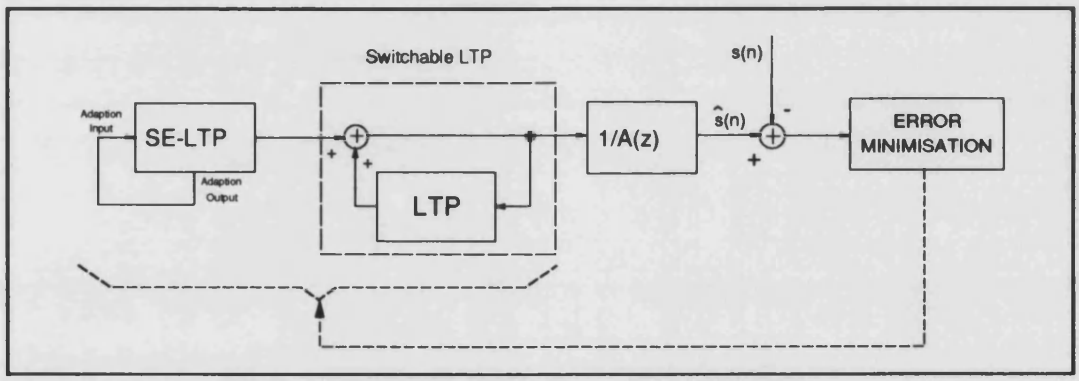

Figure 6.19 *SEV utilising SE-LTP with Switchable Adaption.*

Clear channel objective performance of all three SEVs, Pure, PFC, and SFC, with each adaption method, along with reference coder CELP1G, is shown in table 6.6. The table shows results both without and with incorporation of a LTP. With LTP: Performance of all schemes is approximately equal, there is no longer any advantage with self-excitation. Informal listening tests showed that adaption A produced lower quality speech with the Pure SEV, best illustrated at changes of speaker. Otherwise, the subjective quality of all other SEVs was indistinguishable. With LTP, there is no

significant difference between the initialisation rates of PFC and SFC coders with each adaption. Modifying the adaption has no effect on the attractiveness of the adaptive portion of the codebook. This further reinforces the finding of the previous sections, that the adaptive codebook of the SE-LTP has little advantage over fixed random sequences of a CELP stage, when the LTP is incorporated.

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)	Initialisation Rate (%)
Pure SEV N	6.83 (9.96)	3.60 (3.45)	-
A	6.21 (9.86)	3.90 (3.54)	-
B	6.22 (9.90)	3.92 (3.62)	-
PFC-SEV N	6.67 (10.01)	3.71 (3.54)	28.2 (39.5)
A	6.36 (10.07)	3.68 (3.47)	30.6 (40.2)
B	6.37 (10.08)	3.68 (3.46)	30.3 (40.6)
SFC-SEV N	6.60 (10.01)	3.74 (3.49)	32.9 (48.7)
A	6.31 (10.06)	3.73 (3.46)	38.2 (48.5)
B	6.31 (10.03)	3.72 (3.45)	38.7 (49.9)
CELP1G	5.99 (10.01)	3.66 (3.46)	-

Table 6.6 *Objective Performance and Initialisation Rate of Pure SEV, PFC-SEV, and SFC-SEV, for each Adaption, Values in Parentheses Correspond to Inclusion of the LTP.*

Without LTP: There is a 4-5% drop in the SNRSEG due to the alternative adaptions A and B, however, performance remains superior to the reference CELP scheme. Initialisation rates of the new adaptions A and B were all higher than that of normal adaption. The SEV spends more time operating from the fixed codebook portions and hence as a CELP coder, giving the corresponding drop in performance. When modified adaptions are used, the adaptive portion of the codebook is no longer based upon the exact synthesis filter excitation history, and some of its attractiveness is lost.

Figure 6.20 shows the SNRSEG of all the 10 coders without LTP as a bar chart. The bottom cross-hatched section of each bar indicates the performance of the fully quantised coder, the solid section at the top shows the difference between the fully

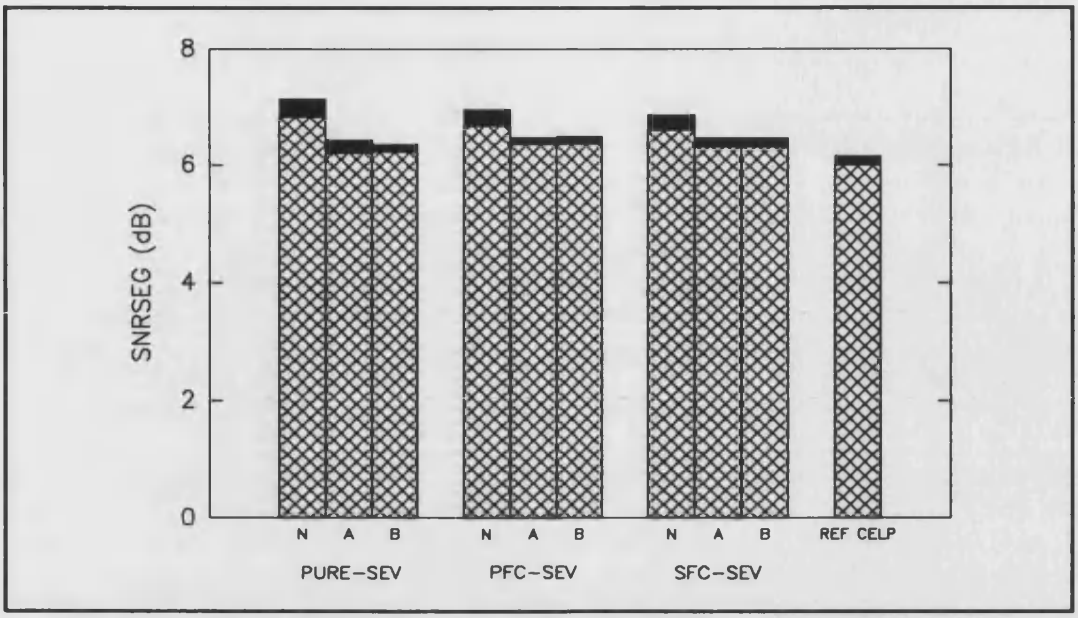

Figure 6.20 *Objective performance of all 3 SEVs without LTP, with each updating strategy.*

quantised coder and the coder with unquantised SE-LTP gain. This clearly shows that the new adaptions A and B have significantly lower gain quantisation loss then the conventional adaption, indicating that the range of unquantised gain values has been reduced. In practice this would allow fewer bits for quantisation of this parameter.

Figures 6.21 to 6.24 are graphs showing the SNRSEG and log-spectral distance against BERs for both PFC and SFC SEVs with and without LTP in random channel errors. On each of the four SNRSEG graphs, the normal adaption curve degrades more significantly than the two modified adaptions. On all graphs there is little difference between the performance of adaptions A and B. With LTP the lower performance of the normal strategy is most pronounced, as the curve for normal adaption crosses the BER axis, the modified adaptions A and B have output SNRSEG of 2dB. Without LTP, this figure is 0.8dB. Subjectively, the normal strategy gave rise to a "bursty" distortion, where a SE-LTP gain term was corrupted in transmission and the synthetic speech was slow to be corrected back to its rightful volume. Where the adaption was independent of transmitted gain, this did not occur and distortion was much more perceptually pleasing.

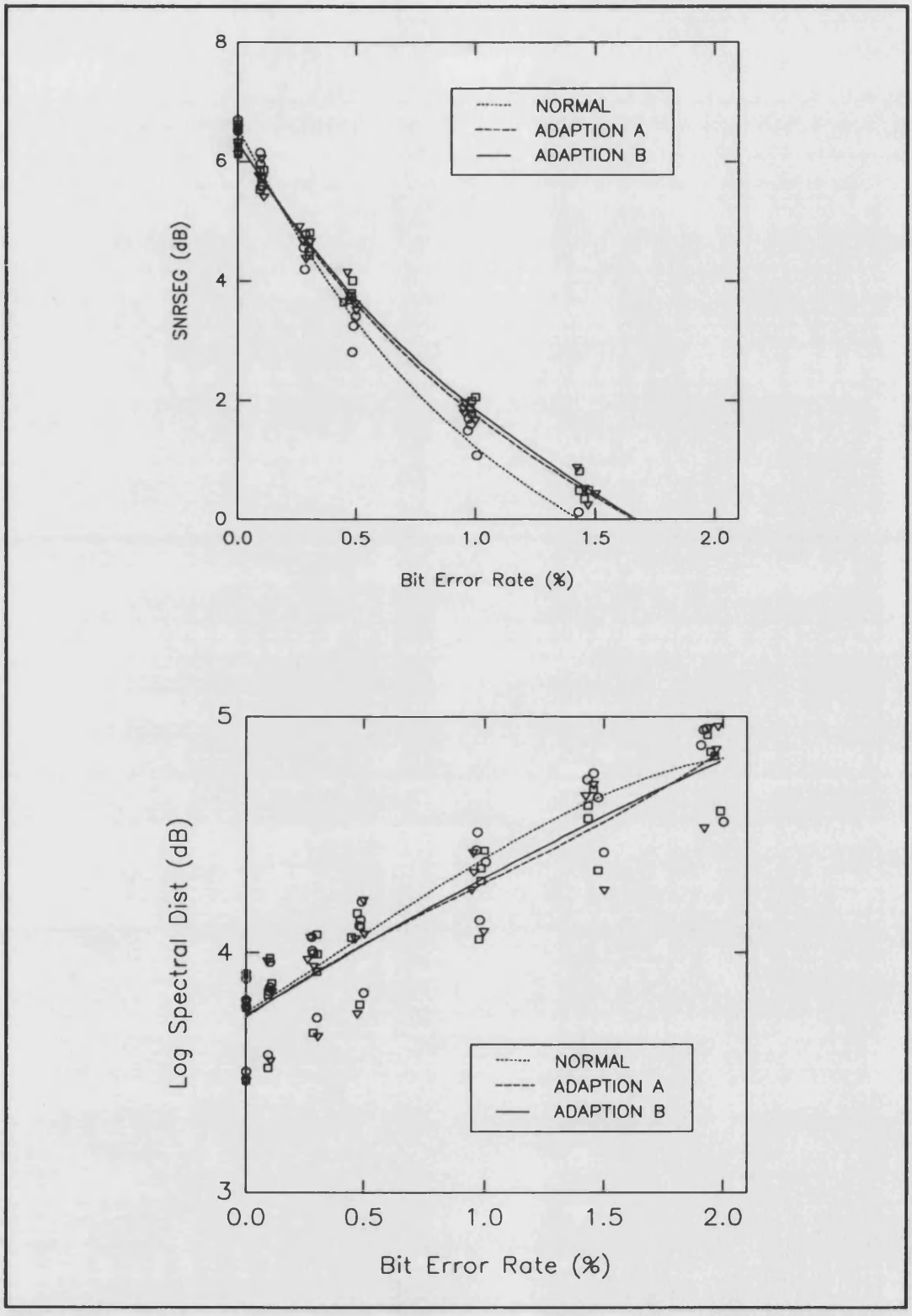

Figure 6.21 *Performance of each Adaption Strategy of the SFC-SEV without LTP in Channel Errors.*

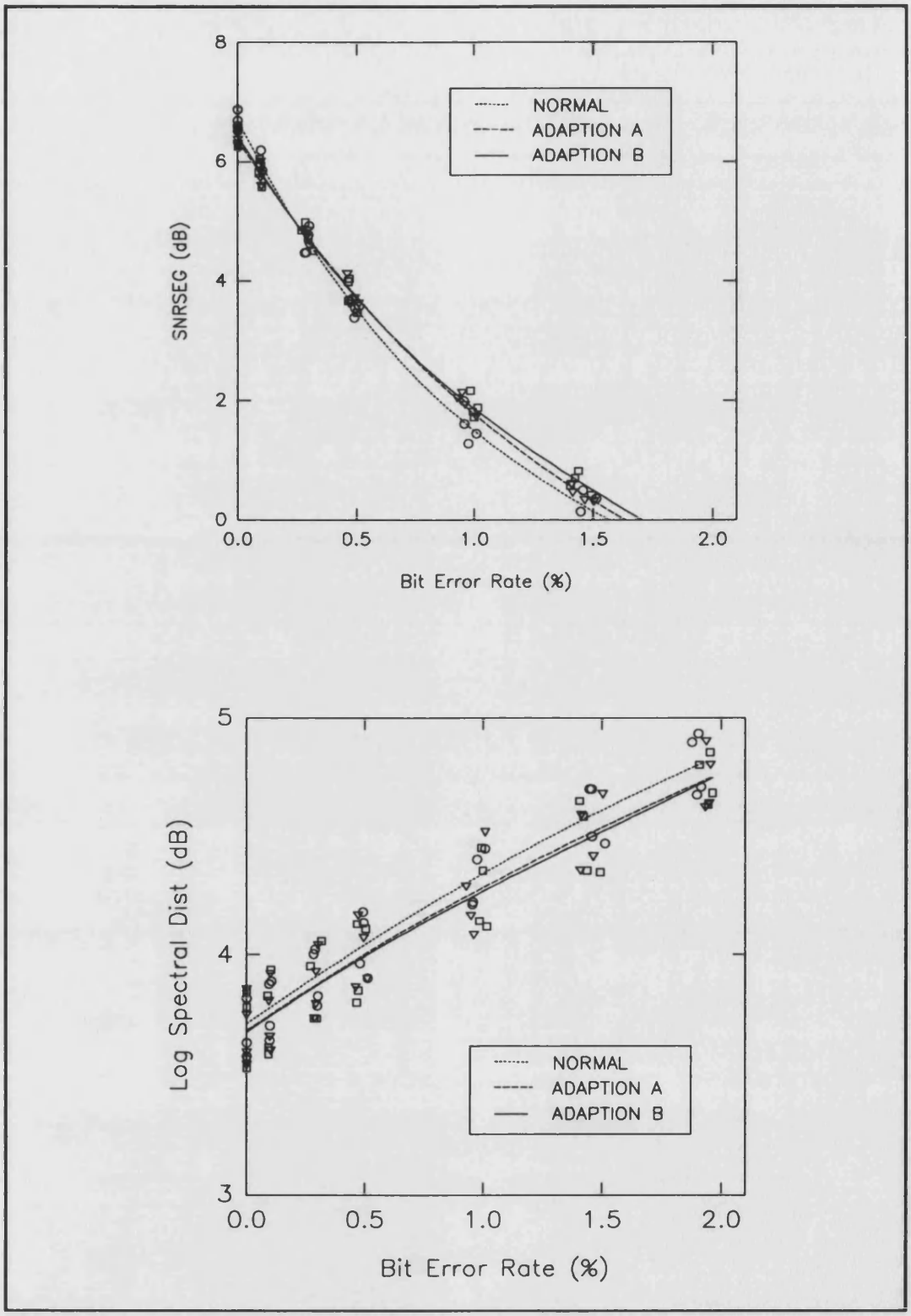

Figure 6.22 *Performance of each Adaption Strategy of the PFC-SEV without LTP in Channel Errors.*

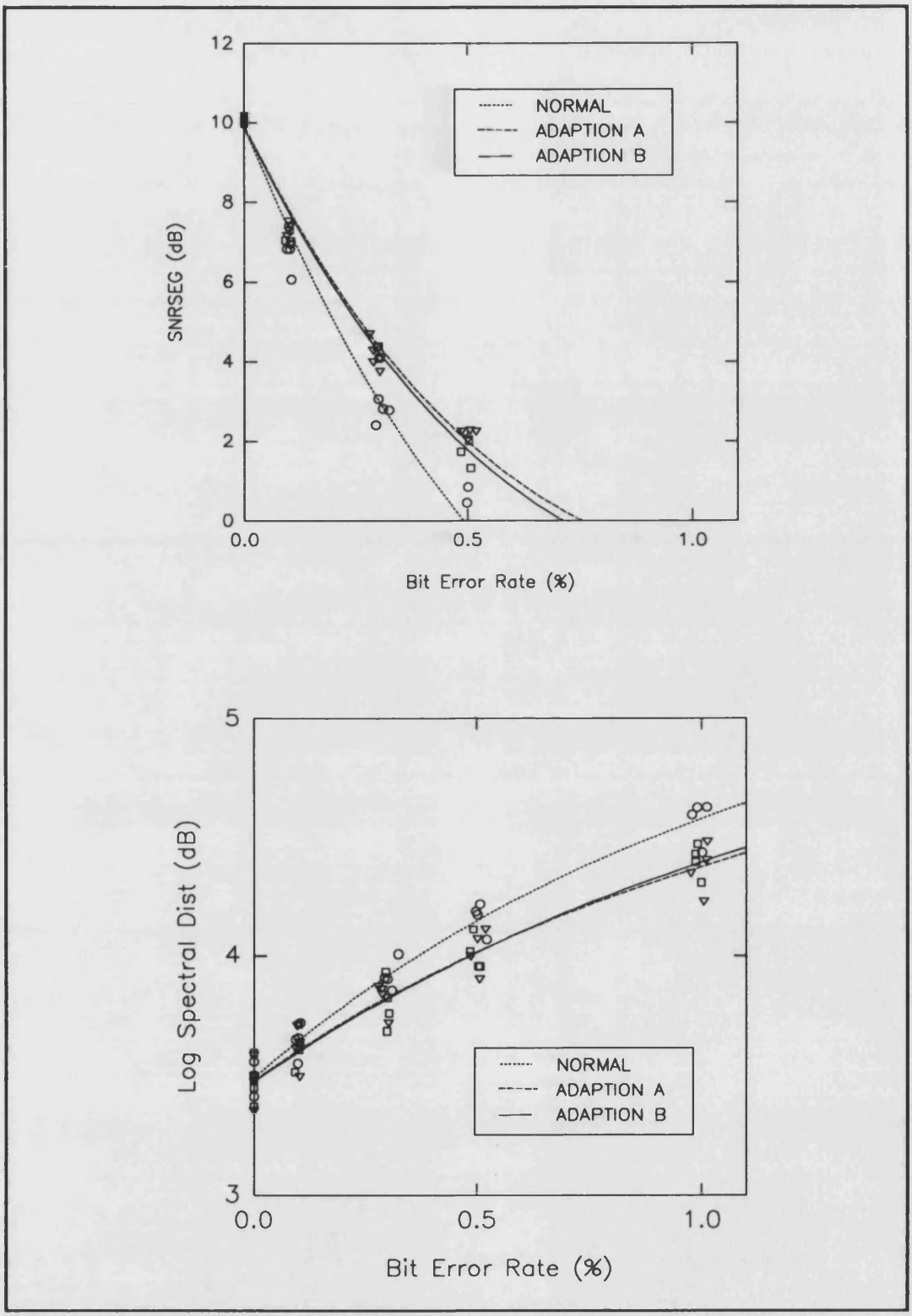

Figure 6.23 *Performance of each Adaption Strategy of the SFC-SEV with LTP in Channel Errors.*

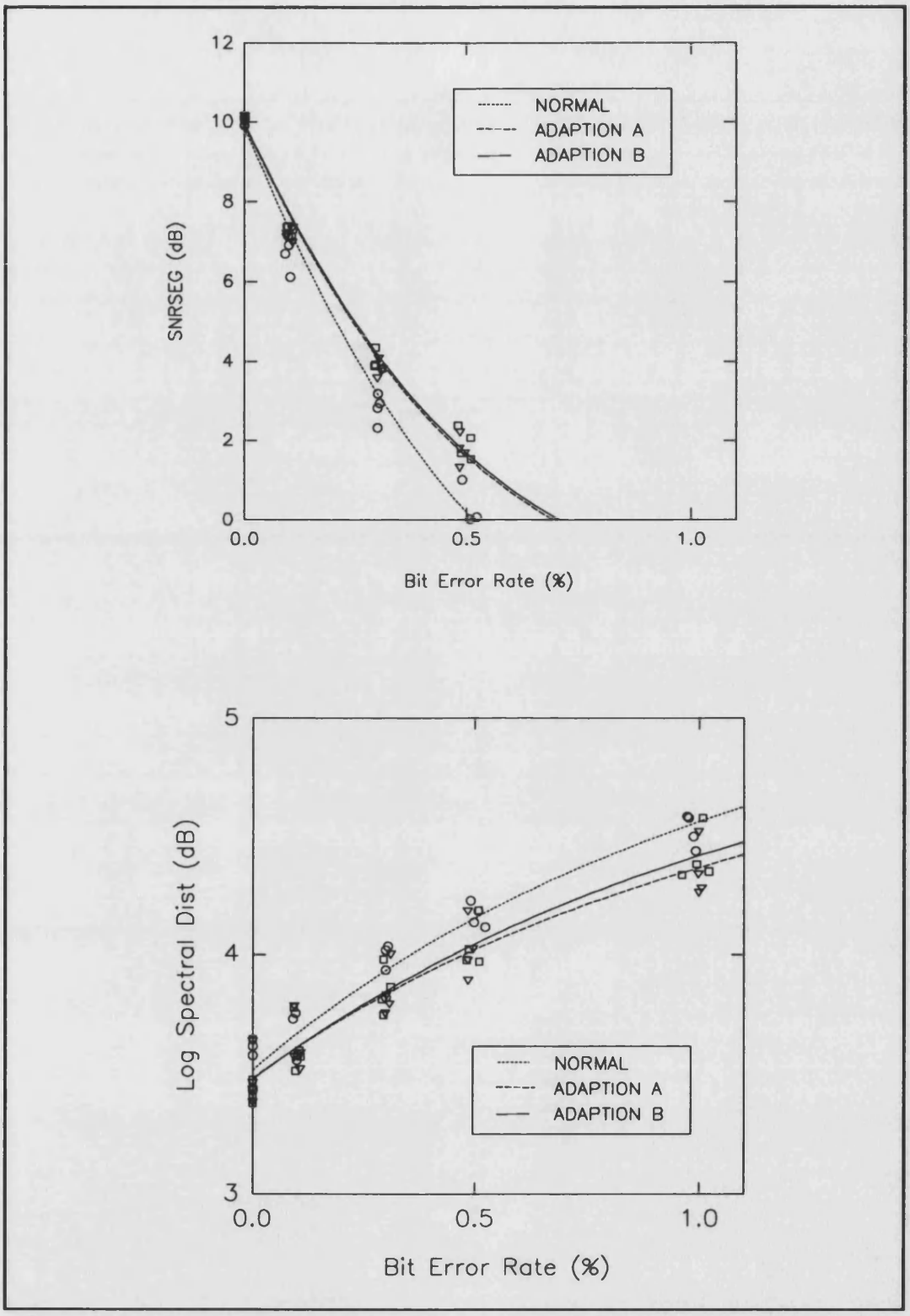

Figure 6.24 *Performance of each Adaption Strategy of the PFC-SEV with LTP in Channel Errors.*

Paired comparison subjective testing was used to compare synthetic speech sentences from the SFC-SEV with all the three adaptions, in both clear channel in figure 6.25 and 0.5% BER in figure 6.26. Each test was performed by 50 listeners hearing two pairs of sentences, one pair from a male speaker and one pair from a female speaker.

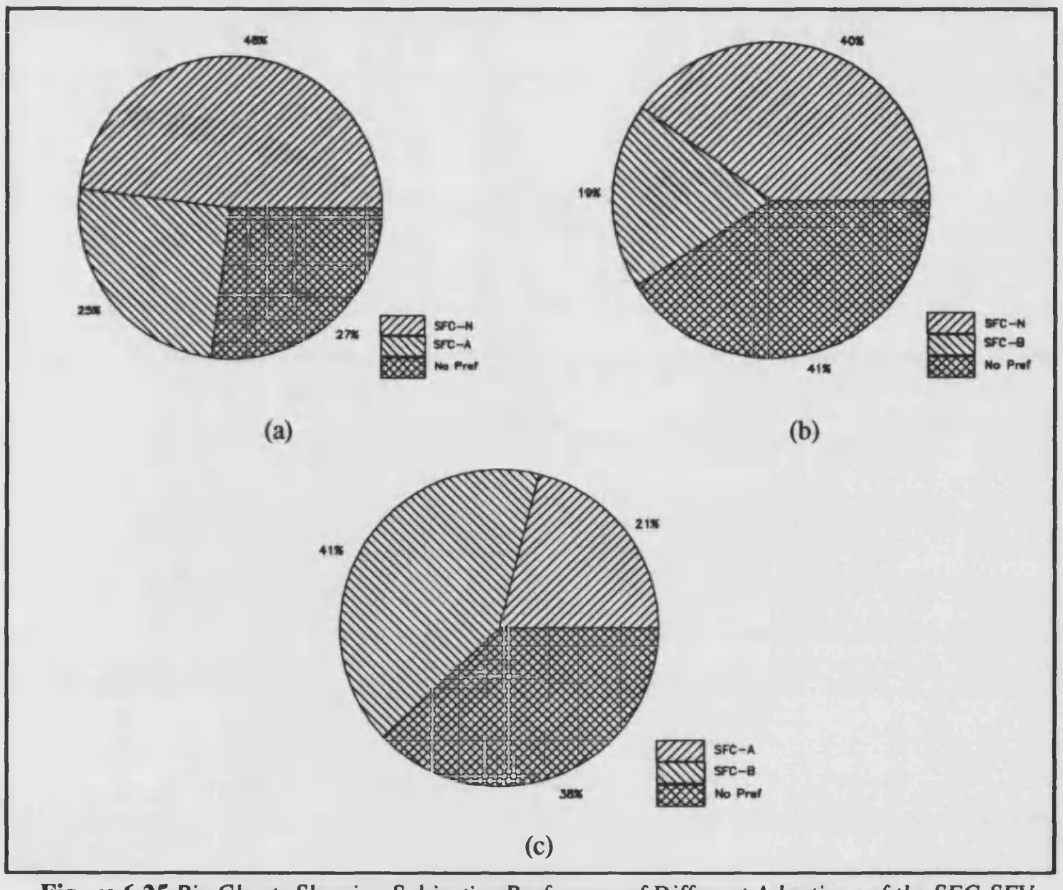

Figure 6.25 *Pie Charts Showing Subjective Preference of Different Adaptions of the SFC-SEV, without LTP, in Clear Channel, (a) Normal Versus Adaption A, (b) Normal Versus Adaption B, (c) Adaption A versus Adaption B.*

The clear channel results are shown in figure 6.25. Pie chart (a), shows normal adaption is clearly preferred to adaption A (48% versus 25%), also from pie chart (b), normal adaption is clearly preferred to adaption B (40% versus 19%). The remaining pie chart (c) compares the two modified adaptions and shows adaption B is clearly preferred to adaption A (41% versus 21%). In clear channel conditions, the order of subjective ranking is normal, adaption B then adaption A.

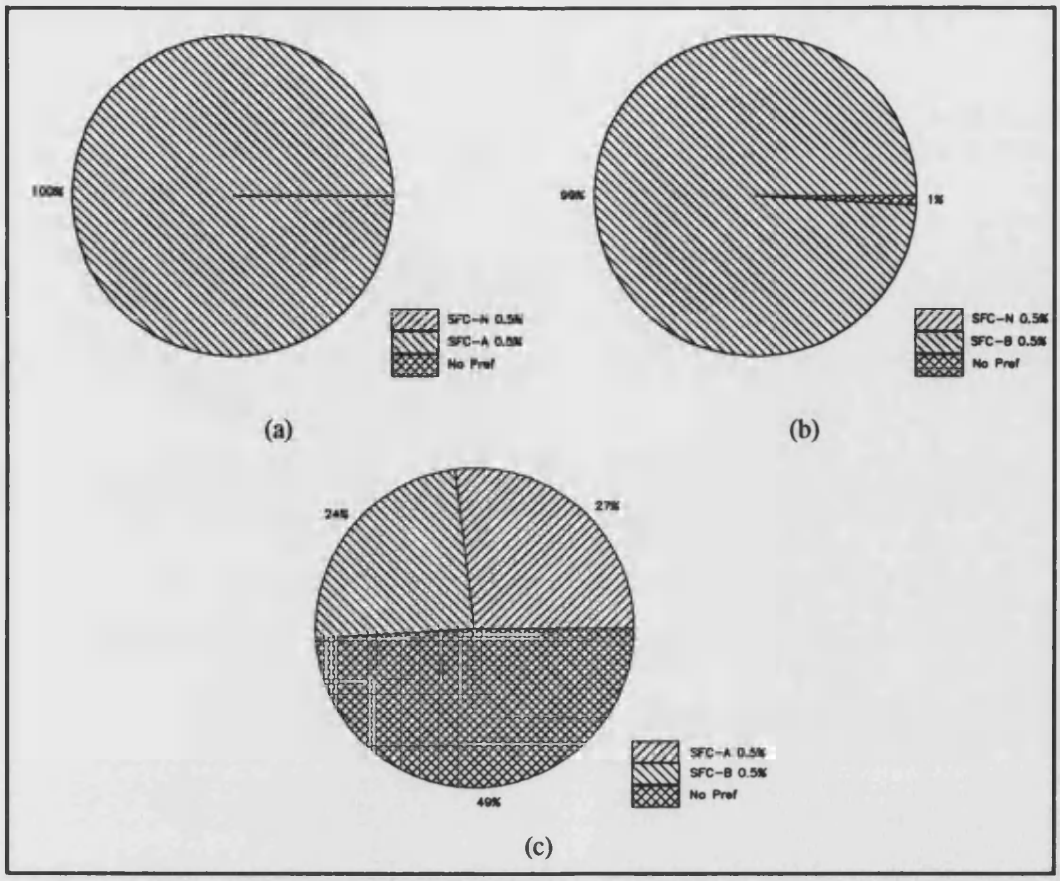

Figure 6.26 *Pie Charts Showing Subjective Preference of Different Adaptions of the SFC-SEV, without LTP, in 0.5% BER, (a) Normal Versus Adaption A, (b) Normal Versus Adaption B, (c) Adaption A versus Adaption B.*

The noisy channel results are shown in figure 6.26, which were taken for a nominal BER of 0.5%. Normal adaption is subjectively totally inferior when compared to either adaption A or B, with the modified adaptions getting 100% and 99% of the selections respectively in pie charts (a) and (b). There was no distinction between outputs of adaptions A and B as shown in pie chart (c) (27% versus 24% with 49% no preferences). Subjectively, the normal strategy gave rise to a "bursty" distortion, where a SE-LTP gain term was corrupted in transmission and the synthetic speech was slow to be corrected back to its rightful volume. With the modified adaptions, this did not occur and the result was a far superior subjective quality.

6.6 Ternary Initialised SEV

The first section of this chapter studied the performance of a number of reference CELP coders and showed that objective and subjective performance was superior when ternary codebooks were used. This section replaces the gaussian initialisation sequence of the SEV with a ternary sequence and studies the resulting performance.

Both the SFC and PFC SEVs had their initialisation sequences replaced with a ternary sequence. Their objective performance was measured and the results shown in table 6.7 for the SFC-SEV and table 6.8 for the PFC-SEV. Without LTP, the difference in SNRSEG is very small in both cases, with increased initialisation rate. With LTP, there is no significant difference between gaussian or ternary codebooks. This very slight improvement is nothing like that seen with the reference CELP coders.

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)	Initialisation Rate (%)
Gaussian	6.60 (10.01)	3.74 (3.49)	32.9 (48.7)
Ternary	6.63 (9.96)	3.61 (3.41)	35.3 (45.9)

Table 6.7 *Objective Performance and Initialisation Rate of SFC-SEV, with Gaussian and Ternary initialisation, Values in Parentheses Correspond to Inclusion of the LTP.*

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)	Initialisation Rate (%)
Gaussian	6.67 (10.01)	3.71 (3.54)	28.2 (39.5)
Ternary	6.72 (10.04)	3.67 (3.46)	32.2 (40.9)

Table 6.8 *Objective Performance and Initialisation Rate of PFC-SEV, with Gaussian and Ternary initialisation, Values in Parentheses Correspond to Inclusion of the LTP.*

The next experiment addressed this apparent lack of improvement in using a ternary initialisation sequence. The ternary sequence has all non-zero elements with magnitude 1. The same is not true of all the values likely to be found in the adaptive codebooks, in practice, they are significantly greater (when the codebook is normally adapted).

This questions the validity of quantising the adaptive vectors and fixed vectors using the same procedure. Utilising the two distinct codebooks of the SFC-SEV, the gain quantisation procedures for fixed and adaptive excitation vectors can be separated. Thus different quantisation tables are used in the SE-LTP depending upon whether an adaptive vector or an initialisation vector forms the excitation.

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)	Initialisation Rate (%)
Normal	6.63 (9.96)	3.61 (3.41)	35.3 (45.9)
Modified	7.03 (10.09)	3.51 (3.45)	40.7 (53.4)

Table 6.9 *Objective Performance and Initialisation Rate of SFC-SEV, with Ternary initialisation, Comparing Normal Gain Quantisation with the Modified Procedure. Values in Parentheses Correspond to Inclusion of the LTP.*

Comparison of the objective performance of the normal gain quantisation procedure with the modified procedure is shown in table 6.9. This shows that with the LTP, there is a negligible improvement however without the LTP output SNRSEG is 5% greater. Objective performance as the BER varies of these two SEVs is shown in figures 6.27 and 6.28. Without LTP, in figure 6.27, the superior clear channel performance of ternary excitation over gaussian excitation remains as the error rate increases and the curves follow very similar gradients. With the LTP, in figure 6.28, the degradation rate is significantly reduced using the ternary initialisation, which is due to the high initialisation rate of 53.4% and is the highest rate seen of any of the SEVs. As a check, this modified gain quantisation procedure was tested with gaussian initialisation and the improvement was found to be negligible.

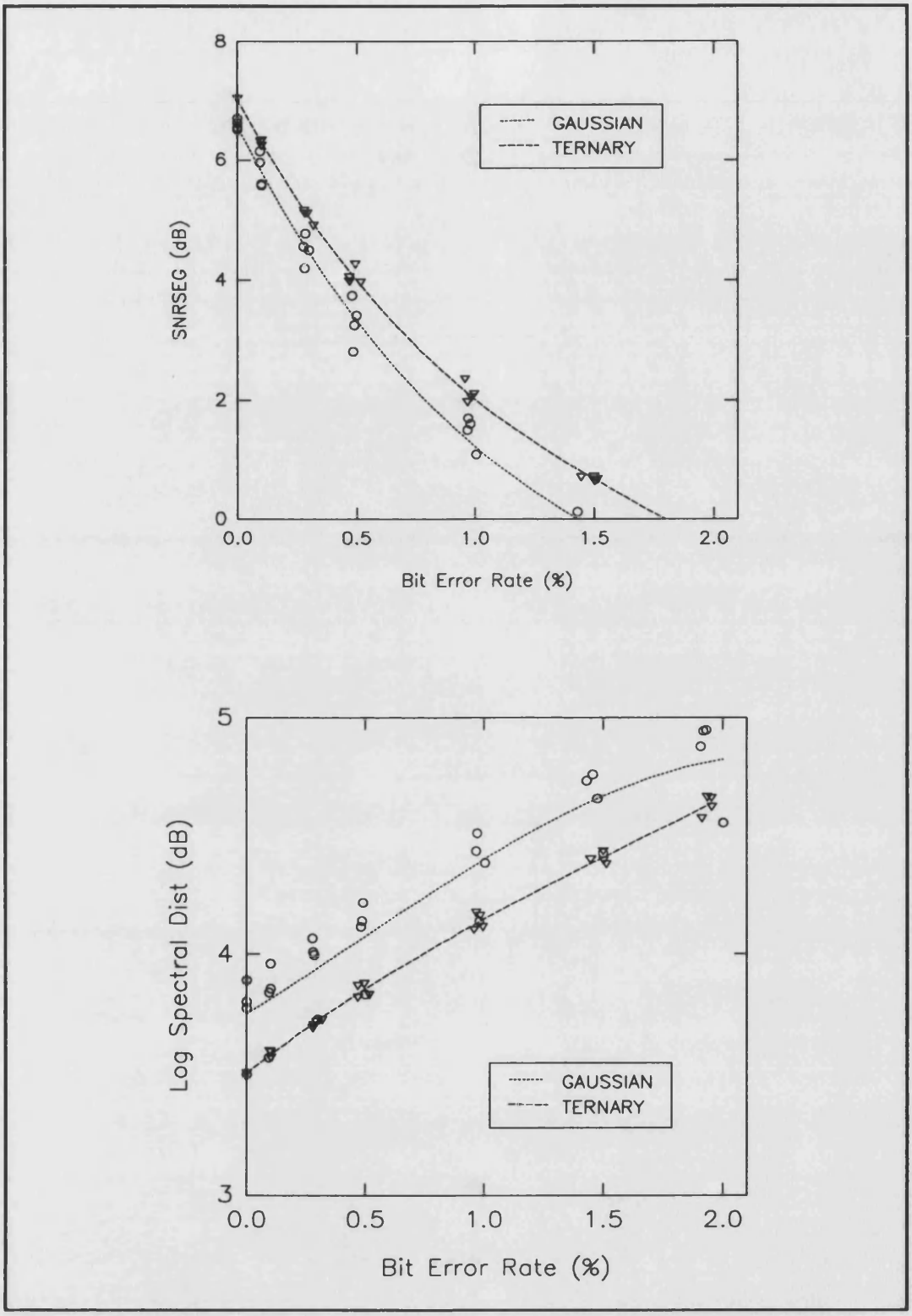

Figure 6.27 *Performance of Ternary Initialised SEV with Modified Gain Quantisation Procedure, Compared to Gaussian Initialisation, without LTP in Channel Errors.*

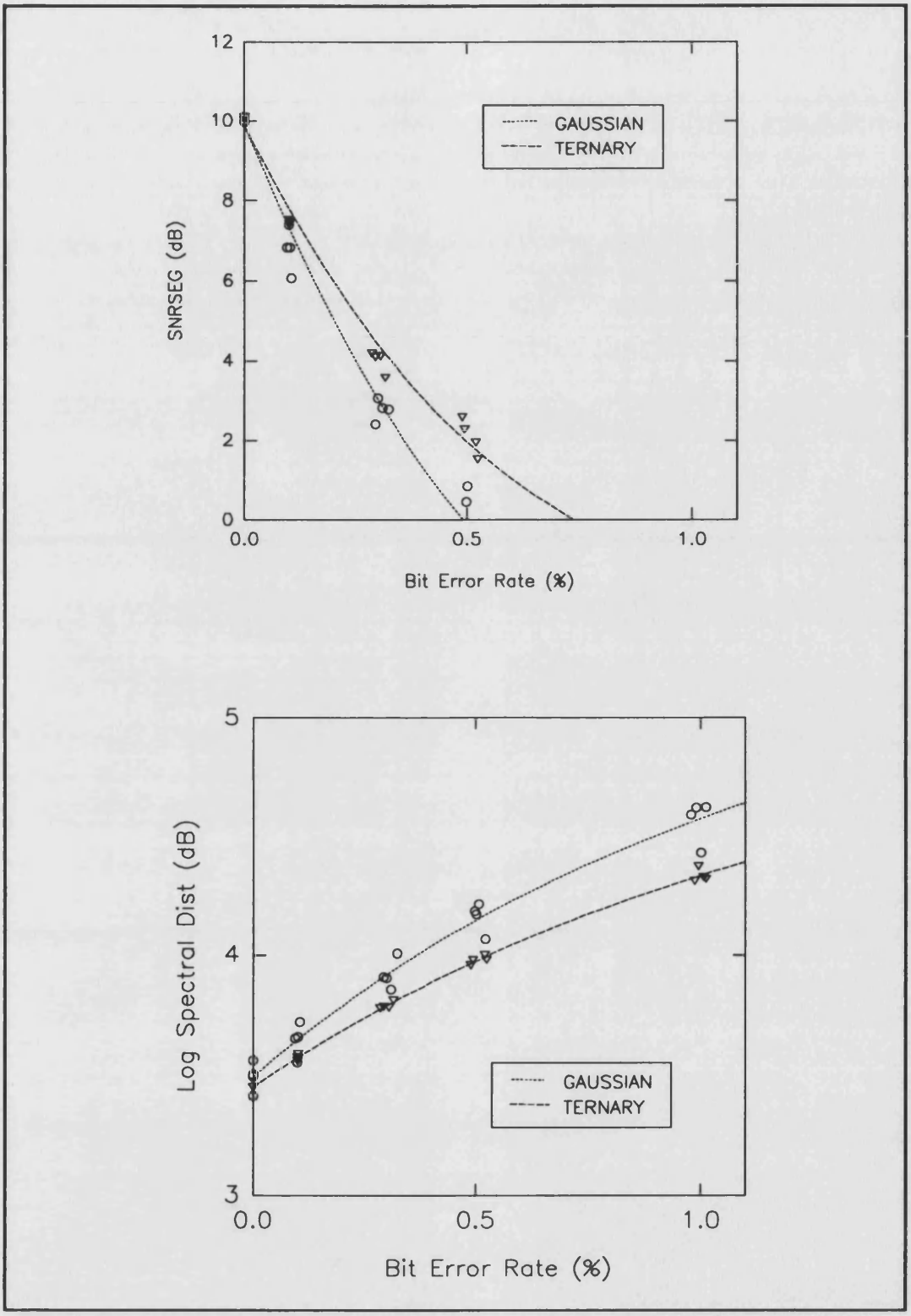

Figure 6.28 *Performance of Ternary Initialised SEV with Modified Gain Quantisation Procedure, Compared to Gaussian Initialisation, with LTP in Channel Errors.*

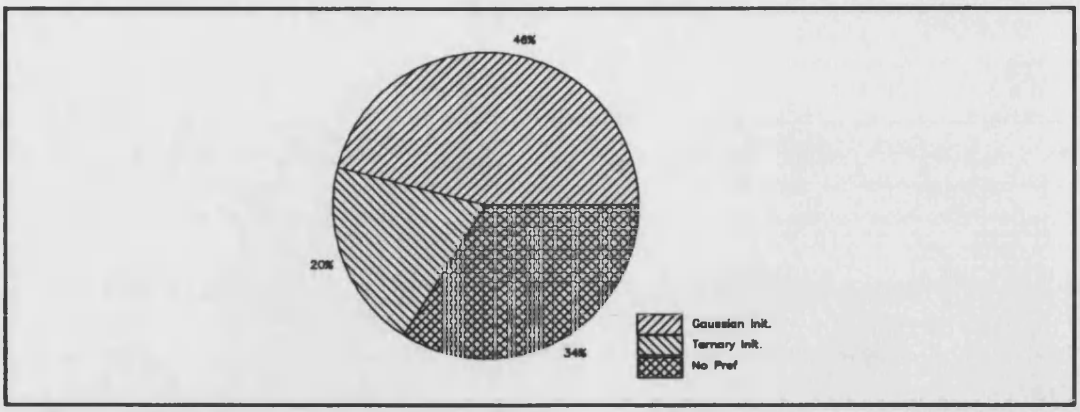

Figure 6.29 *Pie Chart Showing Subjective Preference of Ternary Initialised SEV and Gaussian Initialised SEV.*

A paired comparison subjective test was used to compare the synthetic speech outputs from the gaussian initialised SEV and the ternary initialised SEV (with modified gain quantisation). Both coders did not utilise a LTP. The results are shown in figure 6.29, and show a clear preference for the gaussian initialisation (46% versus 20%). This contradicts the objective measures which showed the ternary initialisation had superior quality. The significant difference between the coders is the initialisation rate, which is 40.7% for ternary initialisation and 32.9% for gaussian initialisation. All the results of this chapter have shown that subjective speech quality is better when self-excitation sequences are used to synthesise speech rather than fixed random sequences. The higher initialisation rate has reduced the subjective speech quality of the SEV.

6.7 Summary

This chapter expanded upon the low complexity SEVs and CELP coders of the chapter 5 and developed a number of fully quantised high complexity coders. An analysis-by-synthesis speech coder module was developed, which could be easily converted into a full speech coder program. All the speech coders described in this chapter were developed from this code section. The aim of the chapter was to develop a Self-Exciting LTP (SE-LTP) stage and to increase its inherent error robustness enabling its use as the basis of a practical speech coding scheme.

Initially, a number of high complexity reference CELP coders were developed. Three variants were developed, two having codebooks with 256 candidate excitation vectors, one with adjacent codebook vectors overlapping for all but one sample and the other overlapping for all but two samples. The third coder had 512 candidate excitation vectors which overlapped for all but two samples. The use of analysis-by-synthesis methods dramatically improved the speech quality, the output SNRSEG was increased threefold to over 6dB. They all had the option of inclusion of a LTP, which again significantly improved the synthetic speech quality, output SNRSEG was increased to over 10dB. Speech quality increased with codebook size. Objective quality was increased with codebook spacing, however this had little effect on subjective quality. Their robustness to channel errors was demonstrated and they were operated with both gaussian and ternary codebooks, which produced different distortions. Paired comparison subjective testing preferred the synthetic speech derived from the ternary codebook.

The chapter then moved on to study the Pure SEV. This coder featured the theoretical SE-LTP where its output was derived entirely from the history of its own previous output. It was initialised at the start of the coding session with a gaussian random sequence. Using analysis-by-synthesis techniques and perceptual error weighting, good speech quality was achieved. Without the incorporation of a LTP, speech quality was far superior to a CELP coder with equivalent sized codebook and the distortion was far more perceptually pleasing. With LTP, speech quality was improved even further, however it was no better than the equivalent reference CELP coder. Although, results were promising in error free transmission, the presence of very slight channel errors lost the identity between adaptive codebooks of encoder and decoder and speech transmission was lost.

The next section introduced two error robust initialisation schemes. They worked by combining fixed and adaptive portions to the SE-LTP codebook. Without any other changes to the SEV, regular initialisation was achieved, with very little loss in clear

channel performance (2-3% of output SNRSEG). Two error robust initialisation schemes were studied, these being the Partial Fixed Codebook (PFC) and Separate Fixed Codebook (SFC) SEVs and performance of both these techniques was found to be very similar. Without LTPs, both these SEVs maintained superior performance to the reference CELP coder and with the LTPs they had equivalent performance. In noisy channels, they showed error robustness, however as the BER increased, they degraded at a faster rate than the reference CELP coder. Paired comparison subjective testing showed unanimous preference for the SEVs over the reference CELP coder when the LTP was not used. However when the LTP was used, the reference CELP coder was more preferable to the PFC-SEV and equally preferable to the SFC-SEV.

The next section introduced two new adaptions for the SE-LTP codebook. Modified adaptions A and B were introduced, they both attempted to constrain the variation in codebook amplitude and further improved the error robustness of the SEV. However, this was achieved at the expense of a small loss in the clear channel performance. In addition the SE-LTP gain quantisation loss was significantly reduced which would enable fewer bits to be used for its quantisation. The noisy channel distortion was perceptually more pleasing, there was no longer the "bursty" distortion typical of corruption of gain terms in conventional adaption. The new adaptions gave the SEV comparable performance to the reference CELP coder in noisy channel conditions.

The final section experimented with ternary initialisation. To get any significant improvement, the gain quantisation had to be separated depending upon whether vectors were selected from adaptive or fixed portions of the codebook. Objective speech quality was improved, however at the expense of significantly increasing the initialisation rate, which reduced the subjective quality. A paired comparison subjective test showed that synthetic speech from a gaussian initialised SEV was preferable to that from a ternary initialised SEV.

Throughout this chapter, results have been listed for coders with and without the LTP. Without LTP, the SEV has always had superior performance to the CELP scheme both objectively and subjectively. This advantage was lost when the LTP was included. To capitalise on this SEV advantage, configurations of SE-LTPs without a LTP must be developed, which is the subject of the next chapter.

Chapter 7

Configurations of SEV

Chapter 7

Configurations of SEV

The previous chapter studied a self-excited vocoder with the excitation source based upon a single self-exciting long-term predictor (SE-LTP). The speech quality was unsatisfactory for a practical scheme. However the superiority of such a scheme over a CELP scheme with an equivalent sized codebook was demonstrated. In addition, a conventional LTP was incorporated in this single predictor scheme, and speech quality was significantly improved. However, this resulted in the losing of the advantage of the SEV stage over the CELP stage. This combination of a SE-LTP with a conventional LTP is termed the "Series SEV". This chapter studies two other SEV configurations, firstly, the "Parallel SEV", which combines two SE-LTP stages in parallel as the excitation source. This gives a significant improvement in the vocoders error robustness, however this is achieved at the expense of some clear channel performance. Secondly, the authors "Series/Parallel SEV" which exploits the clear channel performance of the Series SEV with the error robustness of the Parallel SEV.

Of notable importance within this study is the interrelation between the predictor stages producing the synthesis filter excitation. This distinguishes between the series or parallel nature of the vocoder. This theory is further complicated by the use of the different adaption methods introduced in the previous chapter.

This chapter is divided into 3 sections. Firstly, section 7.1 revisits the Series SEV of the previous chapter, describing the error minimisation procedure and the codebook adaption of predictor stages in more detail. Section 7.2 introduces the Parallel SEV and section 7.3 introduces the Series/Parallel SEV. In this final section, performance all three configurations is compared using both subjective and objective measures. To allow meaningful comparison, the implementations of all three configurations have equal bit rate.

7.1 The Series Configuration Revisited

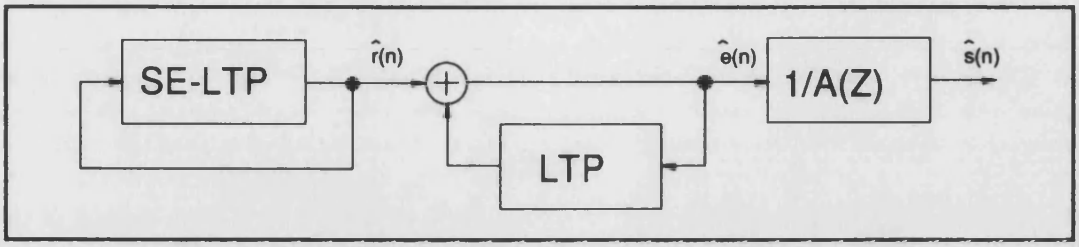

Figure 7.1 *Series SEV Combining a SE-LTP with a Conventional LTP.*

The Series SEV, depicted in figure 7.1, combines a SE-LTP with a conventional LTP to provide the excitation source for the short term synthesis filter $1/A(z)$. The previous chapter was centered upon the SE-LTP unit itself and little attention was given to the interrelation between stages. Equal numbers of bits were used by both predictors for the transmission of gain and delay. Both used 8 bits for delay and 6 bits for gain. The SE-LTP stage had the much larger codebook of both fixed and adaptive vectors and delays were transmitted as integer values, whereas the LTP stage had the much shorter codebook allowing the same number of bits to transmit both integer and non-integer delays. Chapter 6 showed that this configuration produced good speech quality, with output SNRSEG over 10dB. Error robustness was poor with the output SNRSEG reduced to 0dB with 0.5% channel errors. This was improved to 2dB at 0.5% channel errors by using alternative codebook adaptions. With the Series combination, the superior performance of the SE-LTP stage over a fixed codebook stage with equivalent sized codebook was lost.

Each subframe of synthetic speech has a contribution resulting from the current subframe excitation function and from the contents of the synthesis filter memory remaining after synthesis of the previous subframe. From chapter 3, the target speech vector $\tilde{s}(n)$ is given by

$$\tilde{s}(n) = s(n) - \hat{m}(n) \qquad 3.60$$

where $\hat{m}(n)$ is the contribution due to the synthesis filter memory. It is calculated by input of the zero vector to the synthesis filter $1/A(z)$ with initial filter memory set equal to the final filter memory after synthesis of the previous speech subframe.

The synthetic speech output from the Series SEV is given by

$$\hat{s}(n) = \gamma v_l(n - d_l) * h(n) + \zeta v_s(n - d_s) * h(n) + \hat{m}(n) \qquad 7.1$$

where d_l and d_s are the LTP and SE-LTP delays respectively, γ and ζ are the LTP and SE-LTP gains respectively, v_l and v_s are the LTP and SE-LTP codebook buffers respectively, $h(n)$ is the impulse response of the synthesis filter $1/A(z)$ and * denotes convolution and is a memoryless process. Minimisation of the mean squared error gives

$$E = \sum_{n=0}^{N-1} \bar{s}^2(n) - \gamma \sum_{n=0}^{N-1} \bar{s}(n)[v_l(n - d_l) * h(n)] - \zeta \sum_{n=0}^{N-1} \bar{s}(n)[v_s(n - d_s) * h(n)] \qquad 7.2$$

where N is the subframe length. Joint optimisation of the mean squared error for all possible combinations of delay d_l and d_s would present an excessive computational task. In a practical implementation,, they are optimised sequentially and better performance is achieved when the LTP parameters are optimised first.

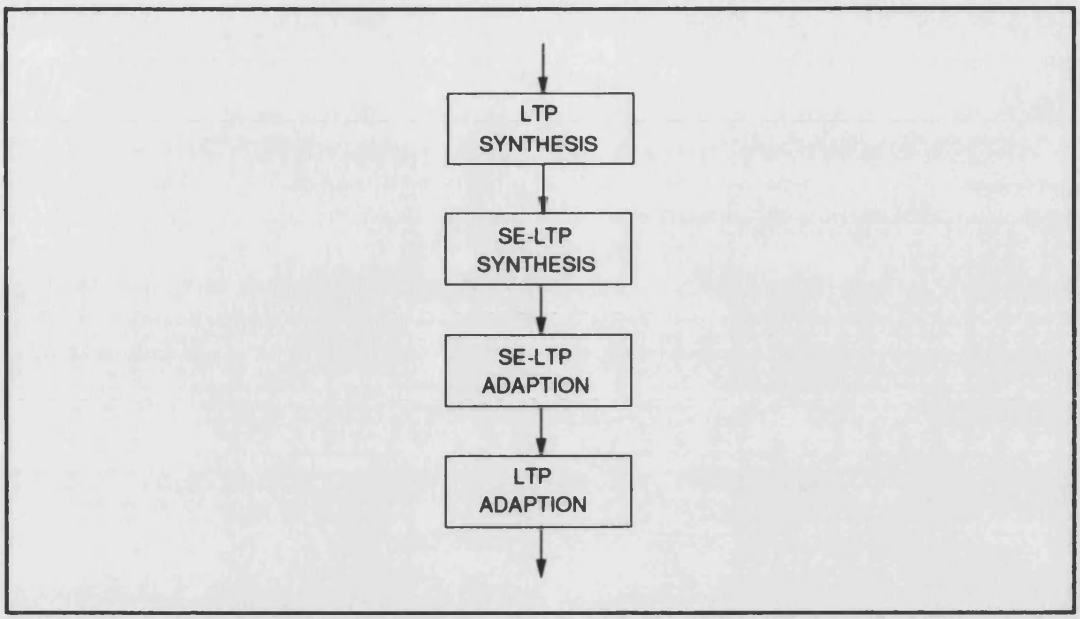

Figure 7.2 *Subframe Operations of the Series SEV.*

A flowchart illustrating the subframe operations of the Series SEV is shown in figure 7.2. The first stage is the optimisation of the LTP parameters. This is performed assuming the contribution from the SE-LTP stage is zero. The LTP synthesis block performs an exhaustive search of its adaptive codebook, over 256 candidate excitation vectors with delays ranging from 20 to 107 samples, 128 being integer values and 128 being non-integer values. A table of all the possible delays is given in appendix 3. Detailed operation of this LTP stage is described in Appendix 1.2.3. The optimum delay d_l maximises $E'(d)$ in equation 3.64. The corresponding gain is then calculated from equation 3.62. This process gives the optimum excitation vector which when convolved with the synthesis filter impulse response gives a synthetic speech vector best fit to the target speech vector $\bar{s}(n)$. The difference between the target vector $\bar{s}(n)$ and this convolution result, is the new target vector for the next stage $\bar{s}'(n)$.

$$\bar{s}'(n) = \bar{s}(n) - \gamma v_l(n - d_l) * h(n) \qquad 7.3$$

The second stage is the optimisation of the SE-LTP parameters d_s and ζ, to give the optimum excitation vector, which when convolved with the synthesis filter impulse response gives lowest error $e(n)$ to the latest target vector $\tilde{s}'(n)$.

$$e(n) = \tilde{s}'(n) - \gamma v_s(n - d_s) * h(n) \qquad 7.4$$

Again an exhaustive search is performed to find optimum delay d_s which maximises $E'(d_s)$ in equation 3.74. The corresponding gain ζ is then calculated from equation 3.72.

The third stage is the adaption of the SE-LTP adaptive codebook. The practical SE-LTP codebook has both fixed and adaptive portions. This section obtains results for both the partial fixed codebook (PFC) SE-LTP and the separate fixed codebook (SFC) SE-LTP, their differences were described in detail chapter 6. Description of the adaption theory will be based upon the SFC SE-LTP.

The SFC SE-LTP codebook is of length 314 samples

$$\{v_{s-i}\} = \{v_{s-314}, v_{s-313}, v_{s-312}, \ldots, v_{s-1}\} \qquad 7.5$$

It contains fixed portions $[v_{s-314}..v_{s-148}]$ and adaptive portions $[v_{s-147}..v_{s-1}]$. The Series SEV operated with analysis framelength of 40 samples. As in the previous chapter, for convenience, the extracted vector used for codebook adaption is placed in the logical continuation of the codebook buffer $[v_{s0}..v_{s39}]$.

Three adaption methods were investigated in chapter 6, Normal, described by equation 6.6,

$$v_s(i) = \zeta v_s(i - d_s) \qquad 0 \le i < 40 \qquad 6.6$$

Adaption A, described by equation 6.11,

$$v_s(i) = v_s(i - d_s) \qquad 0 \le i < 40 \qquad 6.11$$

Adaption B, described by equations 6.12 and 6.13,

$$A = \sqrt{\sum_{k=-314}^{-148} v_s^2(k)} \qquad 6.12$$

$$v_s(i) = v_s(i - d_s) \cdot \frac{A}{\sqrt{\sum_{k=0}^{39} v_s^2(k - d_s)}} \qquad 0 \le i < 40 \qquad 6.13$$

Following one of these three adaptions, the adaption vector is found in the continuation of the codebook buffer $[v_{s0}..v_{s39}]$. The entire codebook is then left shifted to place these elements into the actual adaptive codebook. This is achieved for the SFC SE-LTP by equation 6.9

$$v_s(i) = v_s(40 + i) \qquad -167 \le i < 0 \qquad 6.9$$

It should be noted, that the optimum excitation vector is a multiple of the adaption vector and only in the case of normal adaption are the two equal.

The fourth stage is the adaption of the LTP codebook. In the Series SEV, this is of length 147 samples

$$\{v_{l-i}\} = \{v_{l-147}, v_{l-146}, v_{l-145}, \ldots, v_{l-1}\} \qquad 7.6$$

The adaptive codebook maintains the synthesis filter $1/A(z)$ excitation history, and therefore contains a summation of outputs from both SE-LTP and LTP stages. The excitation vector to be used for the adaption process is equal to the LTP output and is placed in the continuation of the codebook buffer $[v_{l0}..v_{l39}]$. It is given by

$$v_l(i) = \gamma v_l(i - d_l) + \zeta v_s(i - d_s) \qquad 0 \le i < 40 \qquad 7.7$$

The entire codebook is then left shifted by 40 samples.

$$v_l(i) = v_l(40 + i) \qquad -147 \le i < 0 \qquad 7.8$$

The Series SEV was based upon the speech coder module of appendix 1.2 and its bit allocation is shown in table 7.1. Its performance was tested in the previous chapter and is summarised as good clear channel performance, but poor inherent error robustness. The reason being the strict interrelation between the adaption procedures of both

Parameter	Number of Bits
12 LAR Coefficients	54
4 SE-LTP delays	32
4 SE-LTP gains	24
4 LTP delays	32
4 LTP gains	24
Total Bits per 20ms Frame	166

Table 7.1 *Bit Allocation of the Series SEV.*

predictors. The clear channel performance obtained is comparable of that obtained by Salami *et al* [45], however they make no comment about the inherent error performance. The previous chapter noted the "SEV advantage" where the performance of a SE-LTP stage is superior to a CELP predictor stage with equal codebook size. When a LTP is incorporated into the SEV, this advantage is no longer evident. Comparison of the Series SEV with the two other configurations of SEV developed in the next two sections will be presented in section 7.3.

7.2 Parallel Configuration SEV

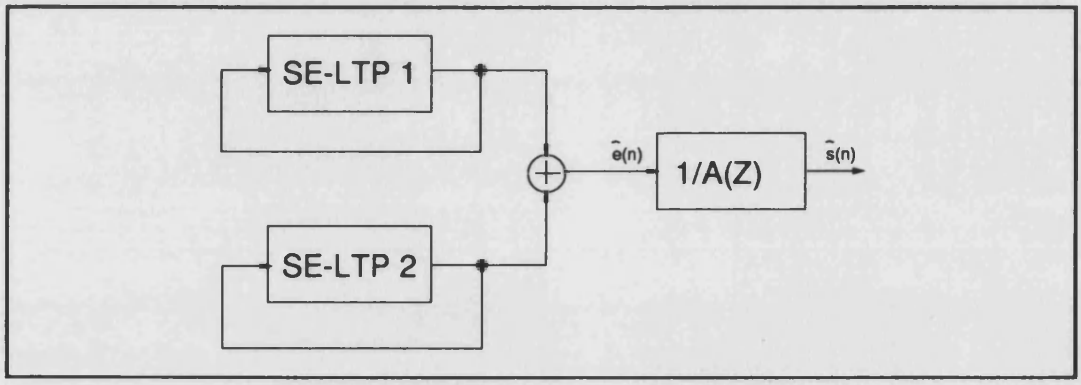

Figure 7.3 *Parallel Configuration of 2 SE-LTPs.*

The Parallel SEV, depicted in figure 7.3, uses two identical SE-LTPs in parallel to provide the excitation for the short term synthesis filter $1/A(z)$. Each predictor operates independently without any interrelation between adaptive codebooks. Both predictors used 6 bits for transmission of gain and 8 bits for transmission of delay. The Parallel

SEV implementation was based upon the speech coder module described in appendix 1. The major difference between the Parallel SEV and the Series SEV code is the lack of a conventional LTP stage and a SE-LTP stage utilising two dimensional arrays giving the two parallel predictors.

The synthetic speech output from the Parallel SEV is given by

$$\hat{s}(n) = \zeta_1 v_{s1}(n - d_{s1}) * h(n) + \zeta_2 v_{s2}(n - d_{s2}) * h(n) + \hat{m}(n) \qquad 7.9$$

where d_{s1} and d_{s2} are the two SE-LTP delays, ζ_1 and ζ_2 are the two SE-LTP gains, v_{s1} and v_{s2} are the two codebook buffers, $h(n)$ is the short term synthesis filter impulse response and $\hat{m}(n)$ is the contribution due to its memory. Minimisation of the mean squared error gives

$$E = \sum_{n=0}^{N-1} \tilde{s}^2(n) - \zeta_1 \sum_{n=0}^{N-1} \tilde{s}(n)\,[v_{s1}(n - d_{s1}) * h(n)] - \zeta_2 \sum_{n=0}^{N-1} \tilde{s}(n)\,[v_{s2}(n - d_{s2}) * h(n)] \qquad 7.10$$

where N is the analysis subframe length. As with the Series SEV, optimisation of parameters for all delays d_{s1} and d_{s2} is unfeasable and in practice, parameters for both stages are optimised sequentially.

Optimisation of gain and delay parameters of the first SE-LTP stage is performed assuming the contribution from the second stage is zero. As with the Series SEV, target vector $\tilde{s}(n)$ is calculated by subtraction of the contribution due to synthesis filter memory. The first SE-LTP synthesis block performs an exhaustive search over all the candidate excitation vectors from its codebook, to find the optimum delay d_{s1} which maximises $E'(d_{s1})$ in equation 3.74. The corresponding gain ζ_1 is then calculated from equation 3.72. This process gives the optimum excitation vector which when convolved with the synthesis filter impulse response gives a synthetic speech vector best fit to the

target speech vector $\bar{s}(n)$. The difference between the target vector $\bar{s}(n)$ and this convolution result, is the new target vector for the next stage $\bar{s}'(n)$. Parameters d_{s2} and ζ_2 from the second stage are then optimised, again according to equations 3.72 to 3.74.

The Parallel SEV has two SE-LTP codebooks, $\{v_{s1-i}\}$ and $\{v_{s2-i}\}$, both are of length 314 samples. Both codebooks are adapted, based upon equations 6.6 and 6.11, 6.12/6.13 from chapter 6. The following set of equations describe the adaption, in each case the symbol x refers to the particular SE-LTP. For Normal Adaption

$$v_{s\mathbf{x}}(i) = \zeta v_{s\mathbf{x}}(i - d_{s\mathbf{x}}) \qquad \begin{matrix} 0 \le i < 40 \\ \mathbf{x} = 1,2 \end{matrix} \qquad 7.11$$

For Adaption A

$$v_{s\mathbf{x}}(i) = v_{s\mathbf{x}}(i - d_{s\mathbf{x}}) \qquad \begin{matrix} 0 \le i < 40 \\ \mathbf{x} = 1,2 \end{matrix} \qquad 7.12$$

For Adaption B

$$A_{\mathbf{x}} = \sqrt{\sum_{k=-314}^{-148} v_{s\mathbf{x}}^2(k)} \qquad \mathbf{x} = 1,2 \qquad 7.13$$

$$v_{s\mathbf{x}}(i) = v_{s\mathbf{x}}(i - d_{s\mathbf{x}}) \cdot \frac{A_{\mathbf{x}}}{\sqrt{\sum_{k=0}^{39} v_{s\mathbf{x}}^2(k - d_{s\mathbf{x}})}} \qquad \begin{matrix} 0 \le i < 40 \\ \mathbf{x} = 1,2 \end{matrix} \qquad 7.14$$

Following one of the three operations, described by equations 7.11 to 7.14, the adaptive portion of both codebooks are left shifted by 40 samples

$$v_{s\mathbf{x}}(i) = v_{s\mathbf{x}}(40 + i) \qquad \begin{matrix} -167 \le i < 0 \\ \mathbf{x} = 1,2 \end{matrix} \qquad 7.15$$

Parameter	Number of Bits
12 LAR Coefficients	54
4 SE-LTP (1) delays	32
4 SE-LTP (1) gains	24
4 SE-LTP (2) delays	32
4 SE-LTP (2) gains	24
Total Bits per 20ms Frame	166

Table 7.2 *Bit Allocation of the Parallel SEV.*

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)	Initialisation Rate (%)
Series N	10.01	3.49	48.7
Parallel N	9.10	3.44	44.7, 34.5
CELP1G	10.01	3.46	-

Table 7.3 *Objective Performance and Initialisation Rates of Series SEV, Parallel SEV, along with Reference Coder (CELP1G).*

Bit allocation for the Parallel SEV is shown in table 7.2, the overall bit rate is equal to that of the Series SEV. The objective performance has been tested and the results are given in table 7.3, along with those of the Series SEV and reference coder CELP1G. The Parallel SEV SNRSEG is lower than that of the Series SEV, (9.10dB versus 10.01dB). This drop is clearly noticeable in informal listening. There is no significant difference in spectral distance between the three coders. The initialisation rate is lower in the Parallel SEV. The SE-LTP with parameters optimised first has lowest initialisation rate of 34.5%, and the other predictor has to 44.7%. Both these figures are lower than the Series SEV, which has initialisation rate of 48.7%. Speech quality improves as the SE-LTP initialisation rate decreases, however, if this rate becomes too low, the error robustness of the vocoder will be undermined.

Lee and Un reported upon a "Multistage Self-Excited Linear Predictive Speech Coder" [29]. This was a Parallel SEV utilising three SE-LTP stages. They report output

SNRSEGs of 4.1dB using a single SE-LTP, 6.1dB using two and 7.6dB using three. The first two of these results are significantly lower than achieved within this thesis and the output SNRSEG of their three stage coder is less than that achieved with two stages in this thesis.

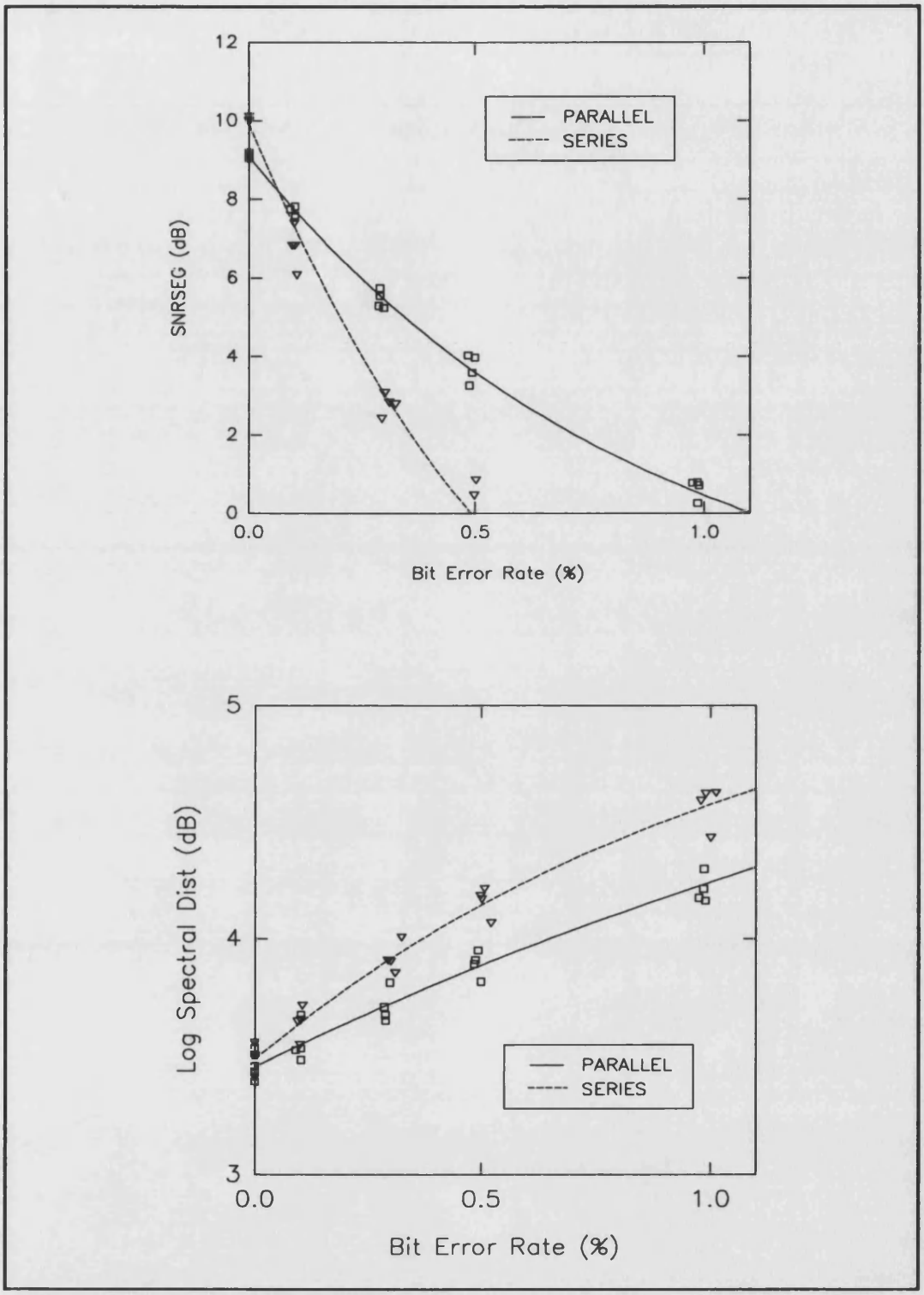

Figure 7.4 *Performance of Parallel SEV constructed from two Separate Fixed Codebook SEVs Compared with the Series SEV, in Channel Errors.*

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)	Initialisation Rate (%)
Parallel N	9.10	3.44	44.7, 34.5
A	9.02	3.45	44.9, 35.8
B	8.80	3.46	46.2, 40.4

Table 7.4 *Objective Performance and Initialisation Rates of the Parallel SEV, for each Adaption.*

Figure 7.4 compares the noisy channel objective performance of both Series and Parallel SEV. The SNRSEG superiority of the Series SEV is short-lived as the BER increases, the Parallel SEV overtaking above 0.1% and degrading at a significantly lower rate. Subjective comparison of Series and Parallel SEVs is performed in the next section.

Objective performance of each coder was also tested with adaption methods A and B, these results are shown in table 7.4. When used, as previous results have indicated, the clear channel output SNRSEG is decreased slightly, by less than 1% with adaption A and by about 3% with adaption B. A better indication of the effects of modified adaption is shown in figure 7.5, which shows the degradation in objective performance with varying BER. Adaption A shows little improvement above the normal method, whereas adaption B has an output SNRSEG 1.5dB greater than the normal adaption at 1% BER. Further subjective testing of both Series and Parallel SEV is covered in the next section.

Performance of the Parallel SEV improves with increasing number of SE-LTPs, hence there is a tradeoff between performance and transmission rate. This could be utilised in digital network applications for reducing network congestion. Bits can be saved by not transmitting bits assigned to the less significant SE-LTPs until the network returns to an uncongested state. Transparent re-initialisation would be achieved within a few subframes, noting that the later optimised SE-LTP stages have a higher initialisation rate, which is beneficial for rapid re-initialisation.

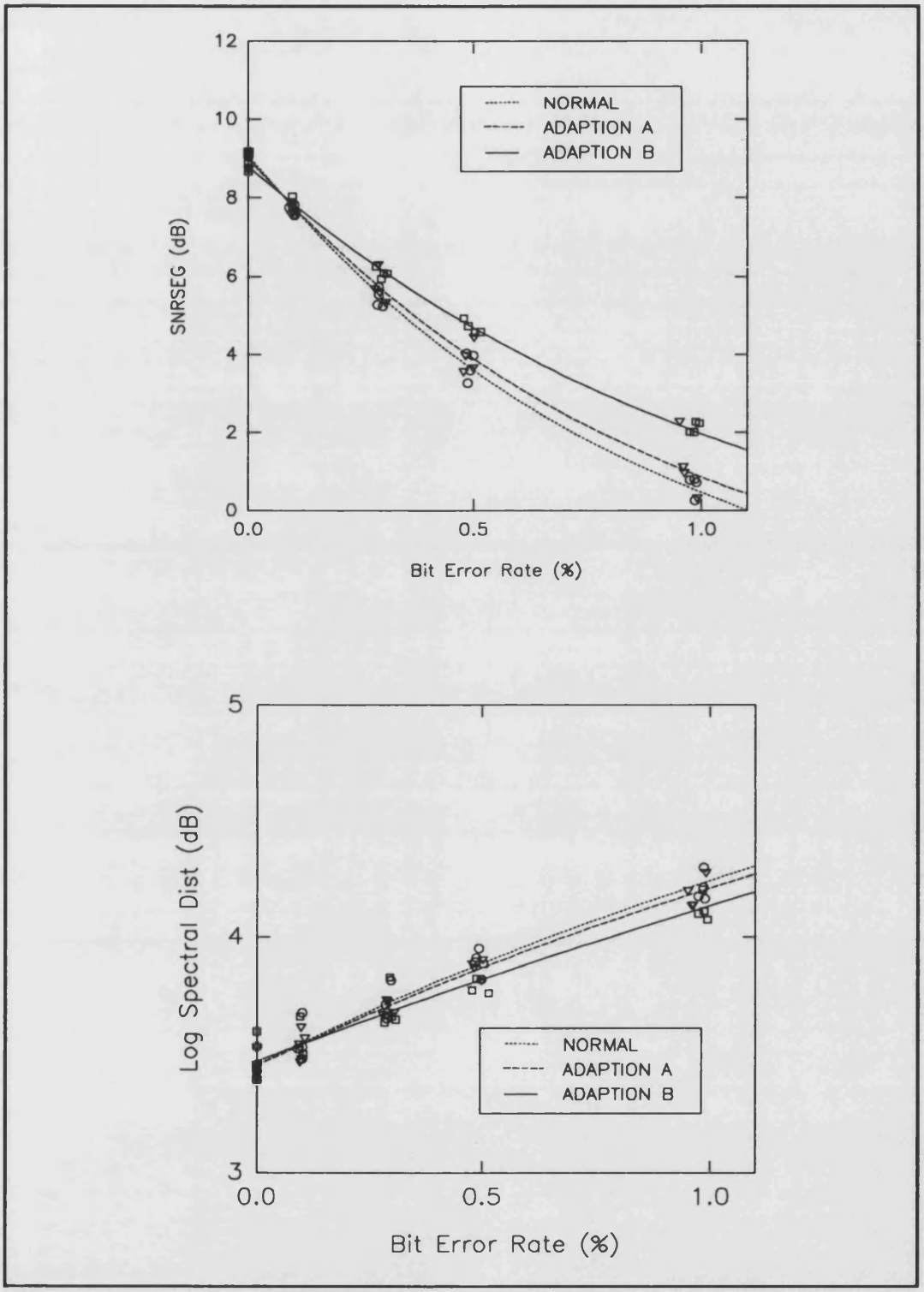

Figure 7.5 *Performance of Parallel SEV constructed from two Separate Fixed Codebook SEVs with each Adaption Strategy, in Channel Errors.*

7.3 Series/Parallel Configuration SEV

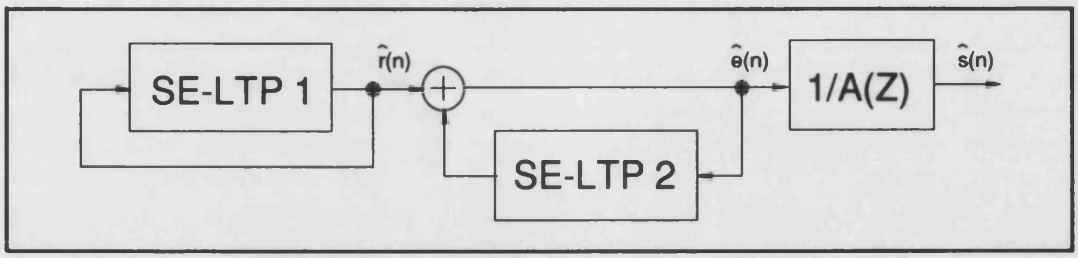

Figure 7.6 *Series/Parallel Combination of 2 SE-LTPs.*

The objective behind the authors Series/Parallel SEV, depicted in figure 7.6, was to keep the error robustness of the Parallel SEV but improve the clear channel performance to approach that of the Series SEV. This was achieved by keeping the two SE-LTP stages of the Parallel SEV, but by connecting them in series with the output of SE-LTP1 forming the input of the SE-LTP2. In operation, this new vocoder operates as a Parallel SEV when SE-LTP2 selects a fixed excitation sequence and it operates as a Series SEV when an adaptive excitation sequence is selected, hence the name "Series/Parallel SEV".

The Series/Parallel SEV shares most of the same code as both the Parallel SEV, the only difference being in the codebook adaption sections. Like the Parallel SEV, output speech is given by equation 7.9

$$\hat{s}(n) = \zeta_1 v_{s1}(n - d_{s1}) * h(n) + \zeta_2 v_{s2}(n - d_{s2}) * h(n) + \hat{m}(n) \qquad 7.9$$

The equation for error minimisation is also identical to that of the Parallel SEV

$$E = \sum_{n=0}^{N-1} \tilde{s}^2(n) - \zeta_1 \sum_{n=0}^{N-1} \tilde{s}(n)[v_{s1}(n - d_{s1}) * h(n)] - \zeta_2 \sum_{n=0}^{N-1} \tilde{s}(n)[v_{s2}(n - d_{s2}) * h(n)] \qquad 7.10$$

Again, parameters for both SE-LTP stages are optimised sequentially, the notable difference from the Parallel SEV is in the codebook adaption stages which more closely resemble those of the Series SEV.

Following one of the three operations, described by equations 7.11 to 7.14, the adaptive portion of the codebook of SE-LTP1 is left shifted by 40 samples. As with the Parallel SEV, this is achieved by

$$v_{s1}(i) = v_{s1}(40+i) \qquad -167 \le i < 0 \qquad 7.15$$

The adaption of the codebook of SE-LTP2 is complicated by the summation with the adaption vector from SE-LTP1.

$$v_{s2}(i) = v_{s2}(40+i) + v_{s1}(40+i) \qquad -39 \le i < 0 \qquad 7.16a$$

$$v_{s2}(i) = v_{s2}(40+i) \qquad -167 \le i \le -40 \qquad 7.16b$$

	SNRSEG (dB)	LOG-SPECTRAL DIST (dB)	Initialisation Rate (%)
Ser/Par N	9.96	3.50	47.8, 27.8
A	9.75	3.49	47.4, 30.8
B	9.13	3.46	46.9, 37.3
Series N	10.01	3.49	48.7
A	10.06	3.46	48.5
B	10.03	3.45	49.9
Parallel N	9.10	3.44	44.7, 34.5
A	9.02	3.45	44.9, 35.8
B	8.80	3.46	46.2, 40.4
CELP1G	10.01	3.46	-

Table 7.5 *Clear Channel Objective Performance and Initialisation Rates of Series/Parallel SEV, Series SEV, and Parallel SEV, for each Adaption.*

Bit allocation for the Series/Parallel SEV is identical to that of the Parallel SEV given in table 7.2. The objective performance of the Series/Parallel SEV was measured, and the results are given in table 7.5, along with those of the Series and Parallel SEVs and reference coder CELP1G. With normal adaption, the Series/Parallel SEV has output SNRSEG of 9.96dB compared to the 10.01dB of the Series SEV, a negligible difference.

Clear channel performance with adaptions A and B is lower as was the case with the Parallel SEV. Output SNRSEG is reduced by 2% with adaption A and 8% with adaption B, these figures are considerably greater than those of the Parallel SEV.

Of all three configurations of SEV, only the Series SEV does not experience a drop in output SNRSEG with the modified adaption procedures, this is further evidence of a SE-LTP stage operating no better than a CELP stage in this configuration. The Series/Parallel initialisation rate is of interest, that of SE-LTP2 being significantly lower than its Parallel SEV counterpart, indicative of a SE-LTP stage usefully modelling utterance pitch. This gives the Series/Parallel SEV its superior speech quality.

Objective performance in varying BERs of all three configurations of SEV with normal adaption, is compared in figure 7.7. The Series/Parallel SEV which has equal clear channel performance to the Series SEV, remains considerably more robust as the BER increases, crossing the BER axis at about 0.9% compared to 0.5% for the Series SEV. At higher BERs, the Parallel SEV has highest SNRSEG, however this is at the expense of much lower quality clear channel performance. Informal listening has shown that the clear channel distortions of the Parallel SEV remain evident as the error rate increases.

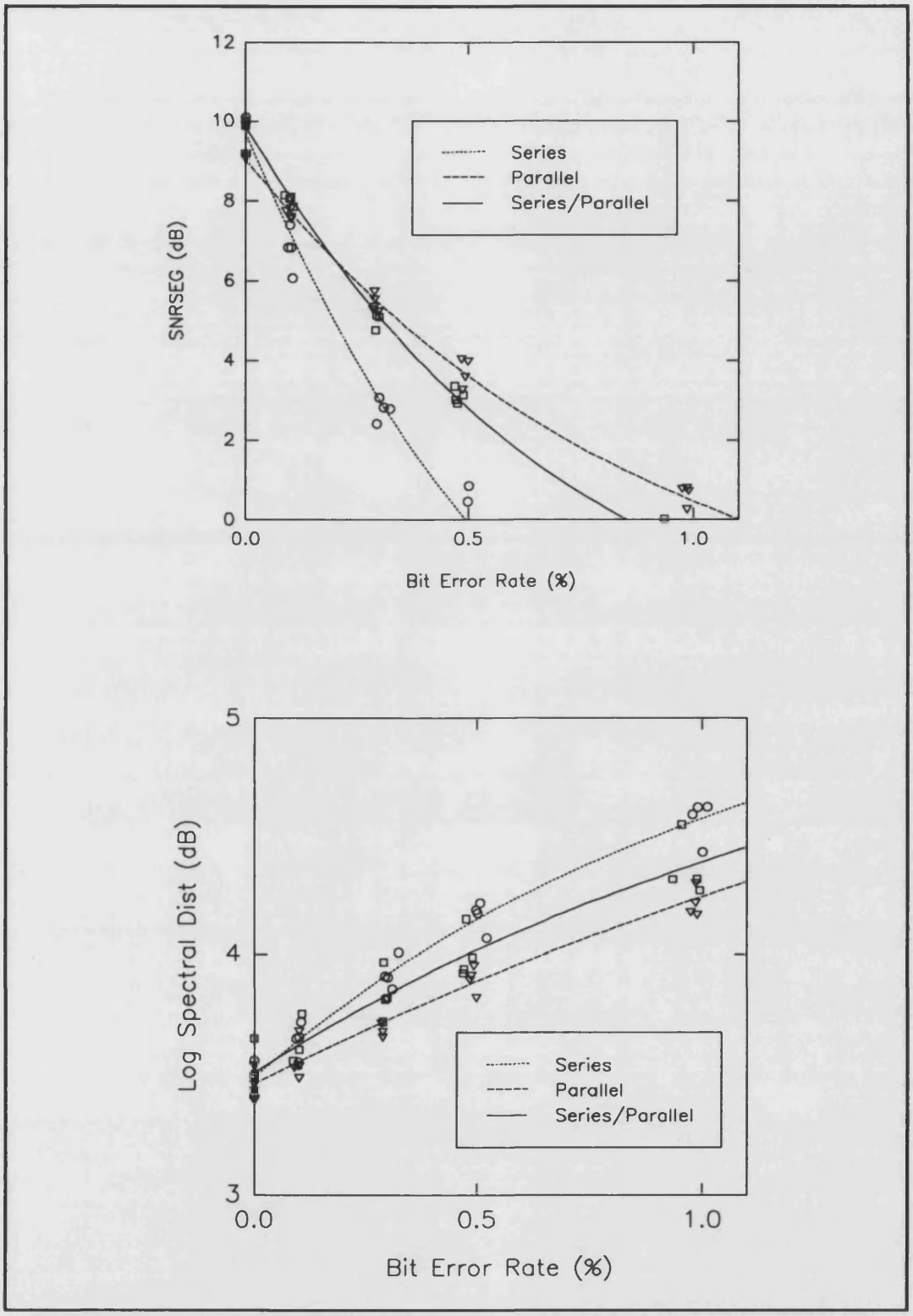

Figure 7.7 *Performance of all three Configurations of SEV with Normal Adaption, In Channel Errors.*

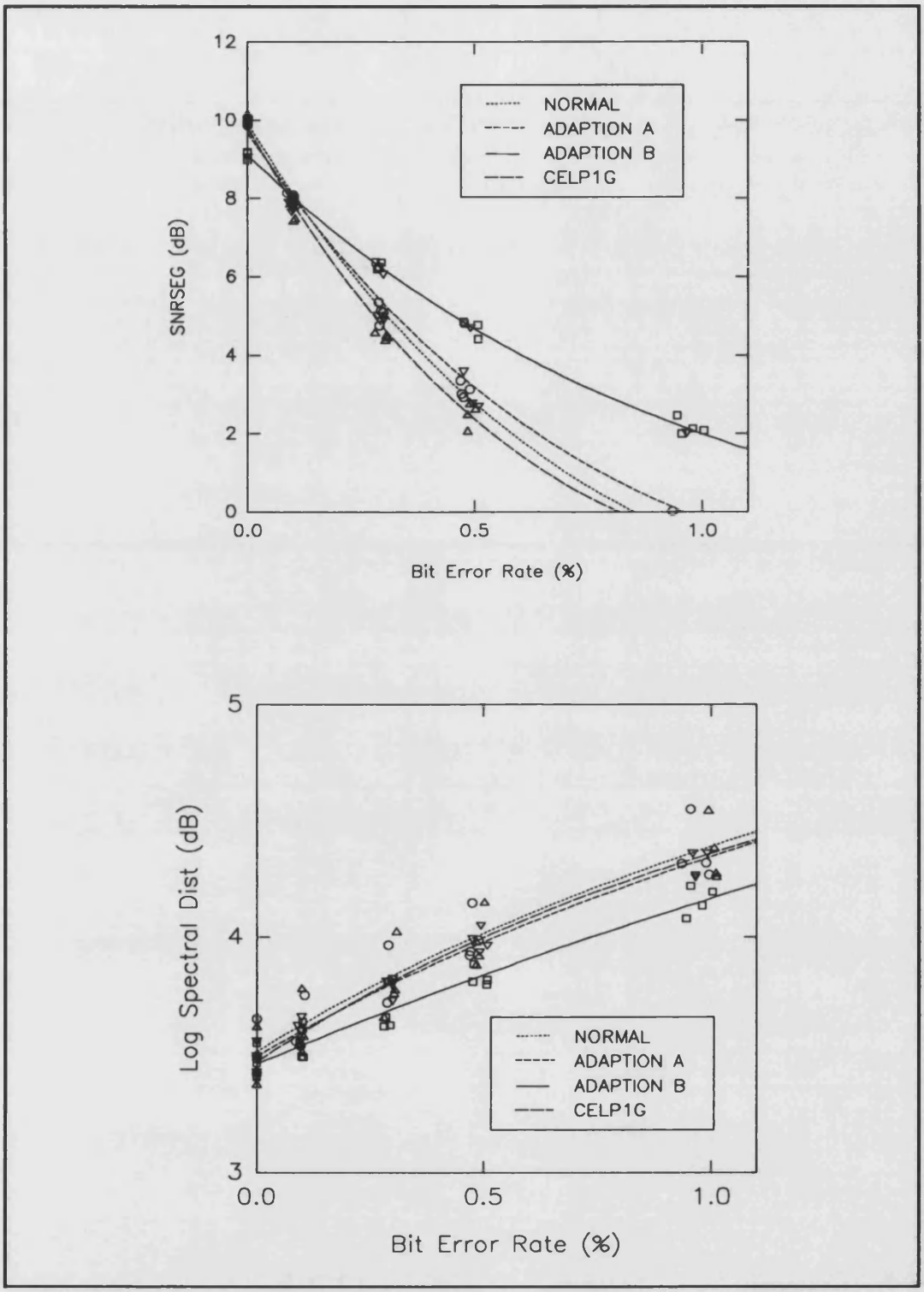

Figure 7.8 *Performance of Series/Parallel SEV with each Update Strategy, Including Reference Coder CELP1G, in Channel Errors.*

Figure 7.8 shows the objective performance of the Series/Parallel SEV with each codebook adaption strategy. Also shown on this graph is reference coder CELP1G. Performance of the normal adaption, adaption A and reference CELP1G is very similar, all having equal clear channel performance and degradation. Adaption B sacrifices some clear channel performance (about 8% of output SNRSEG) to give significantly more robust output at higher BERs, of the order of 3-4dB as BERs of 1% are approached.

The two sets of graphs of figures 7.9 and 7.10 again compare the objective performance of all three configurations of SEV in varying BERs, however this time with adaptions A and B respectively. The performance of adaption A with all the three vocoders was very similar to performance with normal adaption, the only notable difference is an improvement in performance of the Series SEV as the BER increases. From this it is concluded that adaption A performs little better than normal adaption. Whereas with adaption B, the degradation rate has been very significantly reduced with both the Parallel and Series/Parallel SEVs. Both vocoders now have very similar curves, the only difference is slightly higher output SNRSEG (0.33dB) of the Series/Parallel SEV as clear channel conditions are approached.

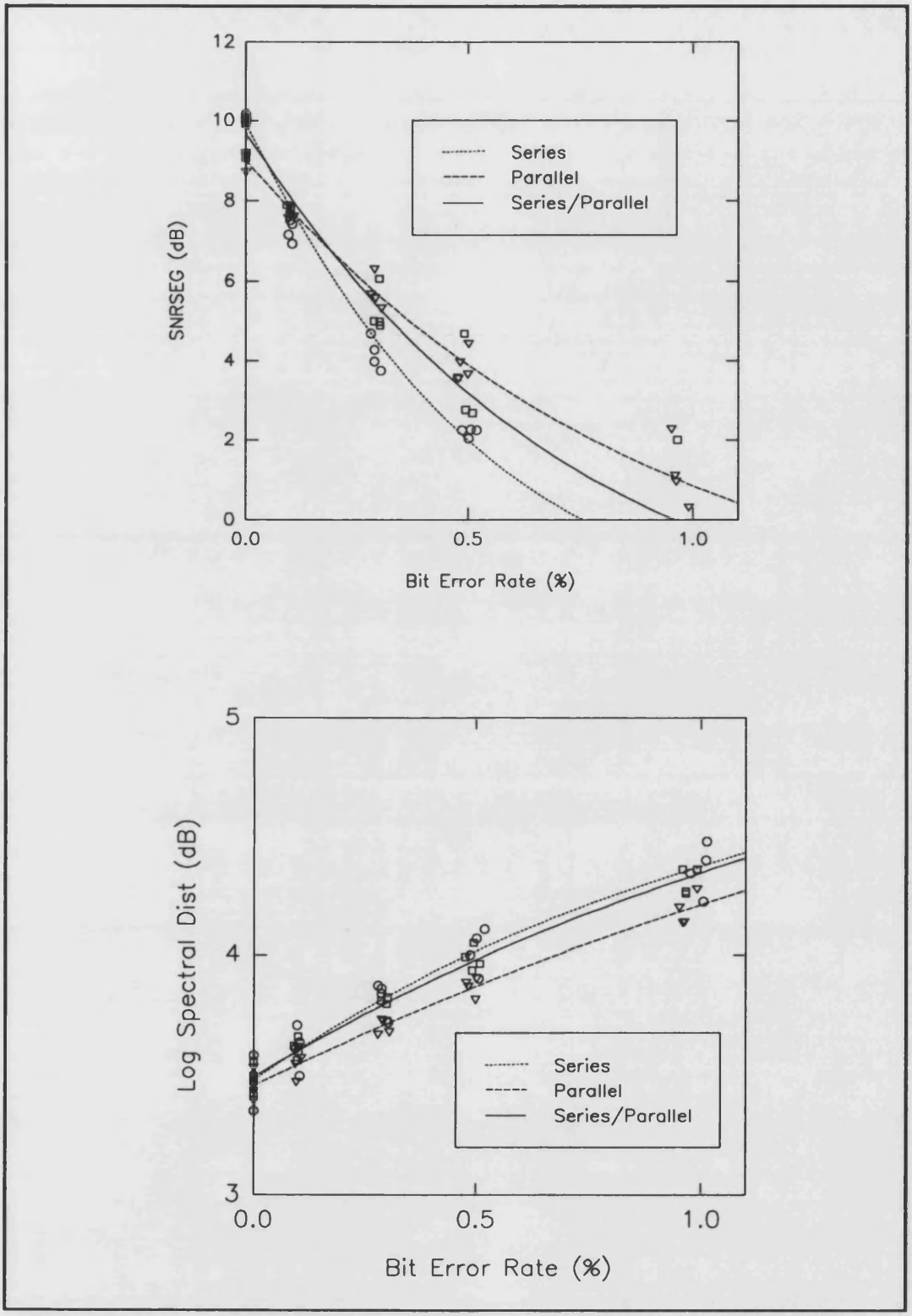

Figure 7.9 *Performance of all three Configurations of SEV with Adaption A, In Channel Errors.*

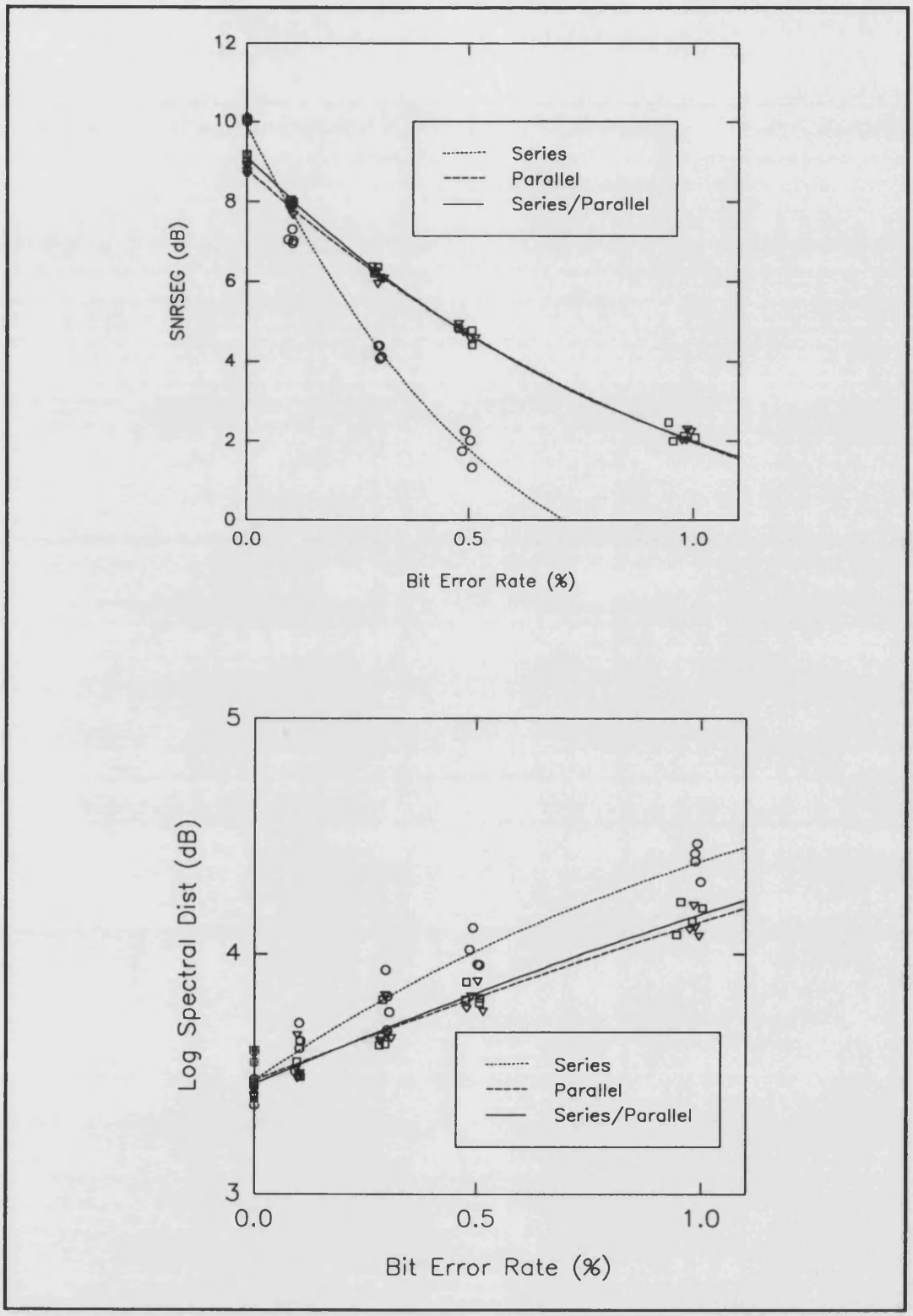

Figure 7.10 *Performance of all three Configurations of SEV with Adaption B, In Channel Errors.*

The remainder of this section describes results obtained from paired comparison subjective testing. Each test consisted of a sentence from the Harvard list of phonetically balanced sentences coded and decoded by two different speech coders and heard side by side. 50 subjects were tested and for each speech coder output, each listener heard both a male and a female speaker, giving 100 paired comparisons for each test. Appendix 4 describes the subjective testing in further detail. The first set of tests compared the performance of all three different configurations of SEV, with normal adaption, both in clear channel and at 0.5% BER. The second set of tests compared the performance of the reference coder CELP1G, with the Series/Parallel SEV with normal and adaption B, again in clear channel and 0.5% BER.

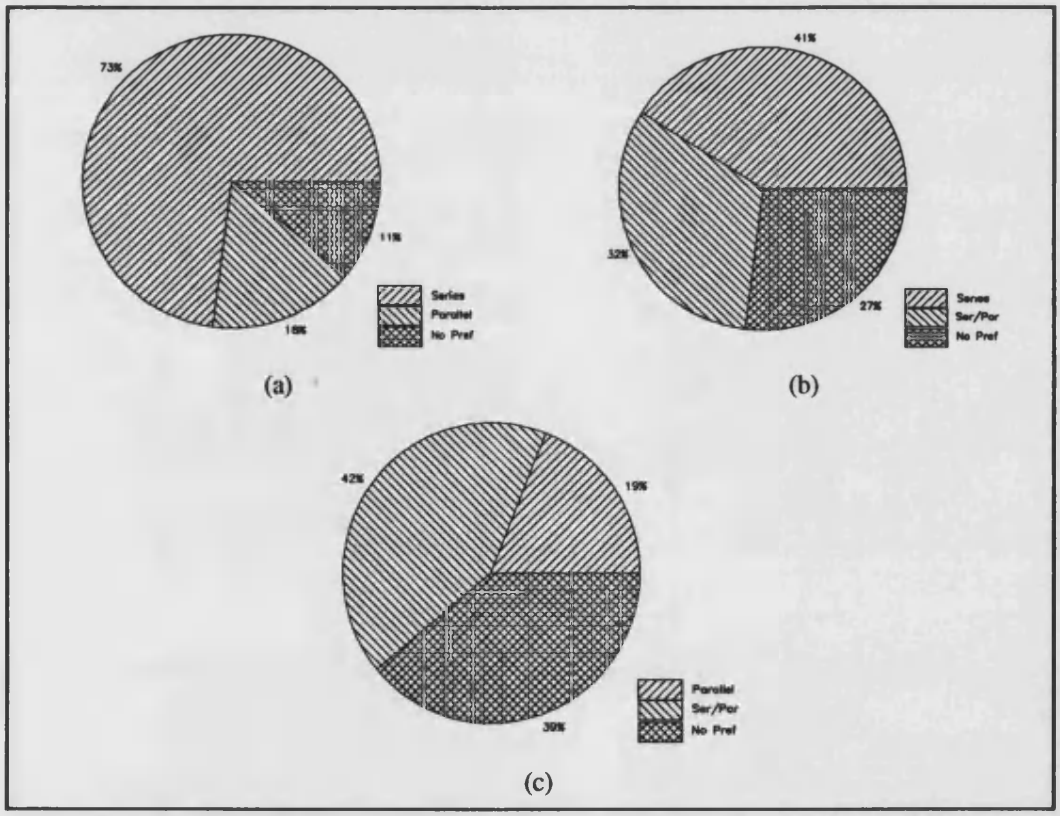

Figure 7.11 *Pie Charts Showing Subjective Preference of Different Configurations of SEV, in Clear Channel, (a) Series Versus Parallel, (b) Series Versus Series/Parallel, (c) Parallel Versus Series/Parallel.*

Figure 7.11 shows the results obtained comparing all three configurations of SEV in clear channel. Pie chart (a) shows the Series SEV totally outclassing the Parallel SEV (73% versus 16%). Whereas comparison of the Series SEV with the Series/Parallel SEV in pie chart (b) is much closer, with the former still maintaining a slight lead (41% versus 32%). The increased number of no preferences is indicative of more similar outputs. Pie chart (c) shows Series/Parallel is clearly preferred to the Parallel SEV (42% versus 19%). In clear channel conditions, the order of preference is Series, Series/Parallel then Parallel which was to be expected from the objective results.

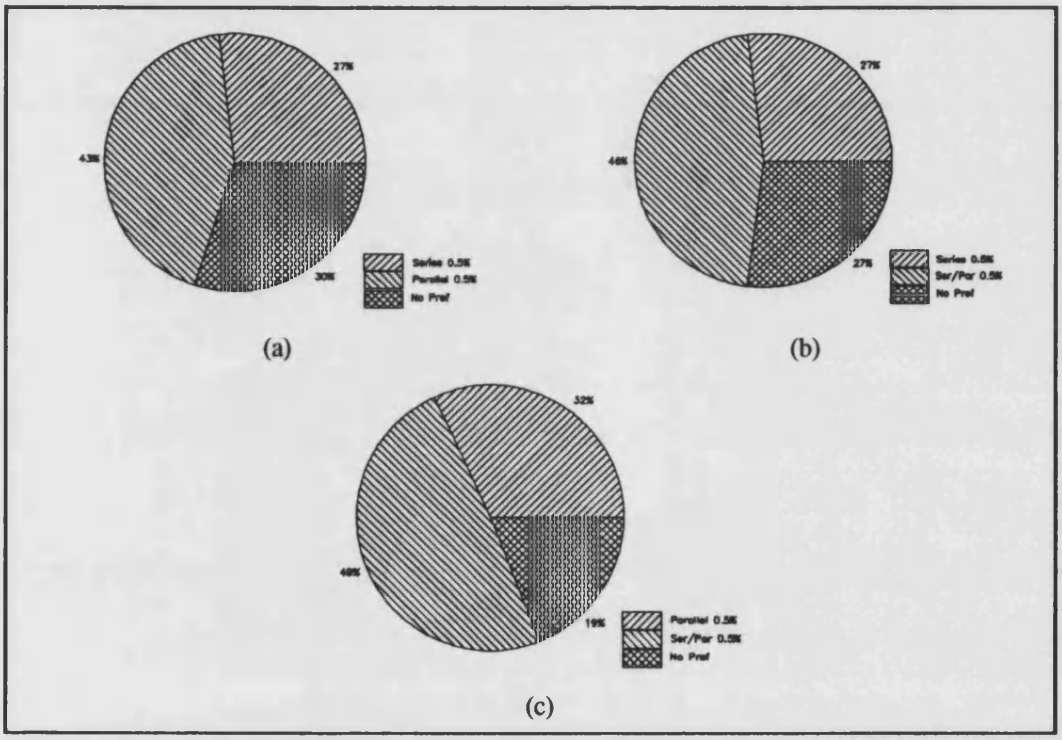

Figure 7.12 *Pie Charts Showing Subjective Preference of Different Configurations of SEV, in 0.5% BER, (a) Series Versus Parallel, (b) Series Versus Series/Parallel, (c) Parallel Versus Series/Parallel.*

Figure 7.12 repeats the above experiment, however this time with a random channel error rate of 0.5%. The subjective preference is now dramatically different as speech coders have subjectively degraded at significantly different rates. Pie chart (a) shows that the Parallel SEV is now preferred to the Series SEV (43% versus 27%). Also from pie chart (b) the Series/Parallel SEV is now also preferred to the Series SEV by a similar

margin (46% versus 27%). Pie chart (c) shows a clear preference for the Series/Parallel SEV over the Parallel SEV (49% versus 32%). In 0.5% BER conditions, the order of preference is Series/Parallel, Parallel then Series. This is not the order expected from the objective measures, it appears that the clear channel imperfections of the Parallel SEV still dominate the subjective quality at a BER of 0.5%.

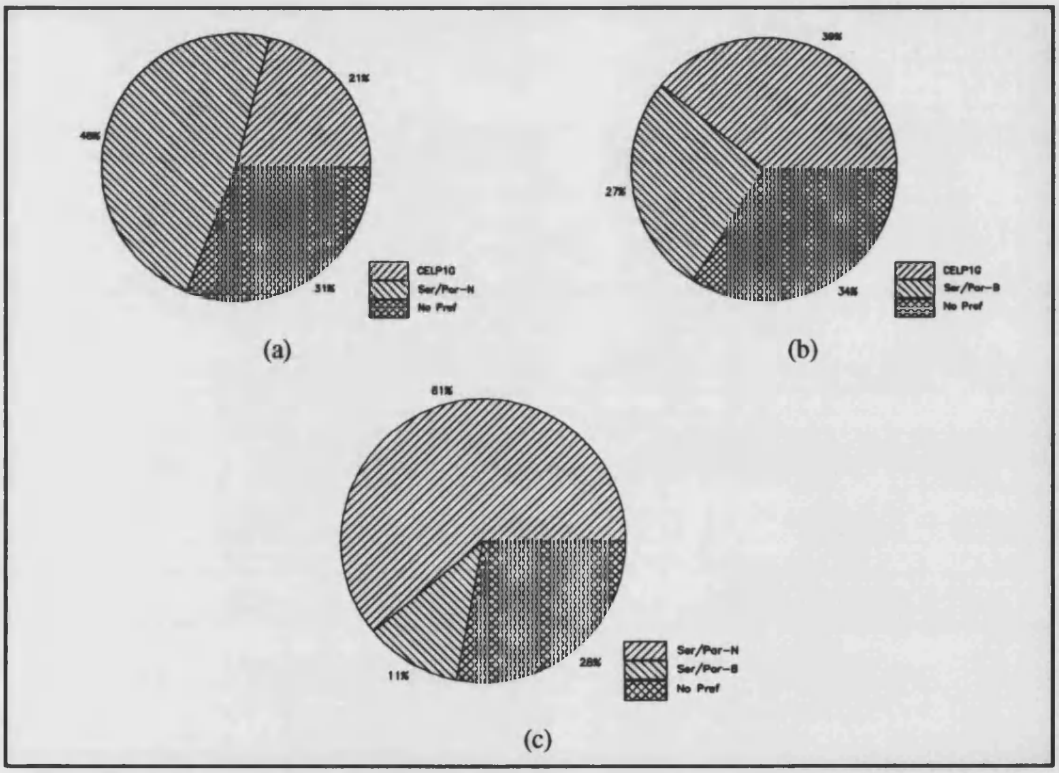

Figure 7.13 *Pie Charts Showing Subjective Preference of Normal Adaption Series/Parallel SEV, Adaption B Series/Parallel SEV and Reference CELP Coder CELP1G, in Clear Channel, (a) CELP1G Versus Series/Parallel-Normal, (b) CELP1G Versus Series/Parallel-B, (c) Series/Parallel Normal Versus B.*

Figure 7.13 shows the results of subjective testing comparing the Normally adapted Series/Parallel SEV, B adapted Series/Parallel SEV and reference CELP coder CELP1G. The normally adapted SEV clearly outperforms the CELP coder as shown in pie chart (a) (48% versus 21%), whereas this domination is lost with adaption B as shown in pie chart (b) (27% versus 39%). Pie chart (c) further illustrates the superior

quality of the normally adapted SEV over the B adapted SEV (61% versus 11%). In clear channel conditions, the order of preference is normally adapted Series/Parallel SEV, CELP1G then B adapted Series/Parallel SEV.

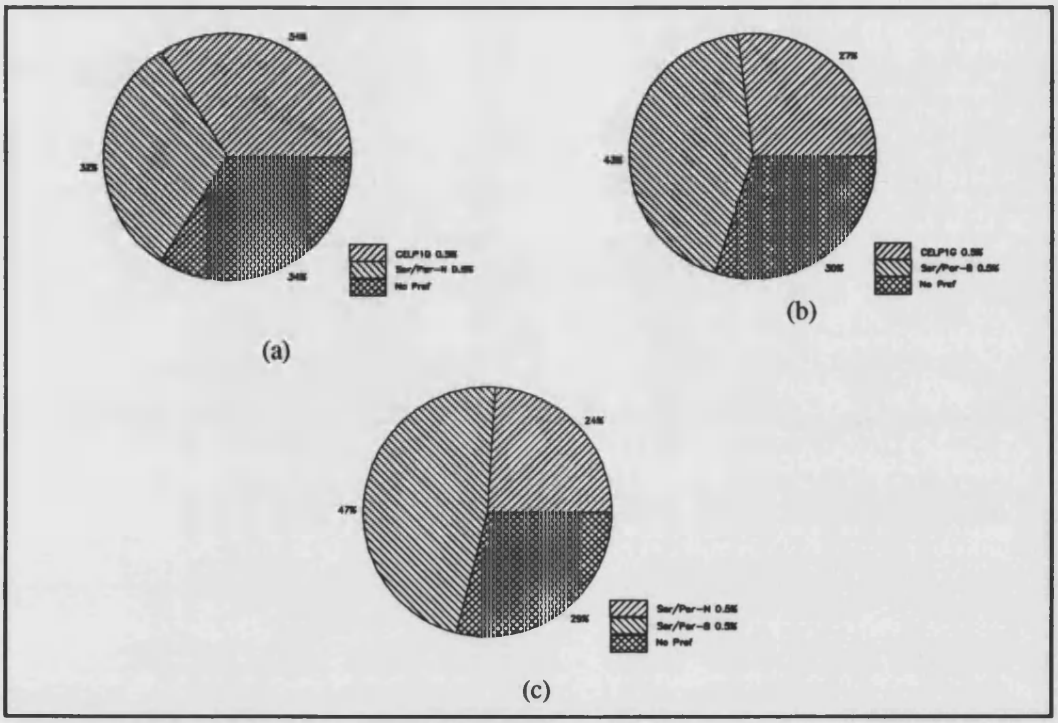

Figure 7.14 *Pie Charts Showing Subjective Preference of Normal Adaption Series/Parallel SEV, Adaption B Series/Parallel SEV and Reference CELP Coder CELP1G, in 0.5% BER, (a) CELP1G Versus Series/Parallel-Normal, (b) CELP1G Versus Series/Parallel-B, (c) Series/Parallel Normal Versus B.*

Figure 7.14 repeats the above experiment, however this time with a random channel error rate of 0.5%. Pie chart (a) shows an equal preference for CELP1G and the normally adapted SEV (34% versus 32% with 34% no preference). Pie chart (b) shows a significantly higher preference for the B adapted SEV over CELP1G (43% versus 27%). Finally pie chart (c) shows a significantly higher preference for the B adapted SEV over the normally adapted SEV (47% versus 24%). In 0.5% BER conditions, the order of preference is the B adapted Series/Parallel SEV over equally placed normally adapted Series Parallel SEV and reference coder CELP1G.

7.4 Summary

A SEV based upon a single Self-Exciting long-term predictor (SE-LTP) has superior speech quality than a CELP scheme with an equivalent sized codebook, however, it is below that required for a practical mobile telephony scheme. Using the SE-LTP and the LTP as two building blocks, this chapter investigated three configurations of SEV with a view to improving the synthetic speech quality.

The improvement in performance from the incorporation of a conventional LTP into this single SE-LTP vocoder was studied in the previous chapter. For the purposes of this chapter, this combination is termed the Series SEV. This chapter revisited this combination, detailing the error minimisation procedure, and the codebook adaption more thoroughly. The previous chapter showed this configuration produced good synthetic speech quality with poor error robustness. Speech quality was aided by the ability of the LTP stage to model non-integer delays.

Attention was the turned to the Parallel SEV, which combined two predictors without any strict interrelation between their adaptive codebooks. To achieve this, both had to be SE-LTPs, operating in parallel with their outputs summed before input to the synthesis filter. Output SNRSEG was reduced by about 9%, with the robustness to channel errors dramatically improved. Robustness was further improved by using modified adaptions A and particularly B at the expense of a small drop in clear channel performance.

Attention was then turned to the authors Series/Parallel SEV, which aimed to keep the error robustness of the Parallel SEV along with the clear channel performance of the Series SEV. This was also achieved by two SE-LTP stages, however, unlike the Parallel SEV, the output of SE-LTP1 was input to SE-LTP2. When SE-LTP2 selected an initialisation sequence, the SEV operated as a Parallel SEV and when SE-LTP2 selected

an adaptive sequence, the SEV operated as a Series SEV. As a result, this vocoder shared most of the code from the Parallel SEV, however the codebook adaption resembled that of the Series SEV.

The three different configurations of twin predictor SEVs were compared in detail using both objective and subjective measures. The Series/Parallel SEV had equal clear channel objective performance to the Series SEV (both had output SNRSEG of 10.0dB) and paired comparison subjective testing placed it slightly behind (41% preference versus 32%). This situation was dramatically reversed in a noisy channel with a nominal BER of 0.5% when the Series/Parallel SEV had significantly greater output SNRSEG of 2.8dB compared to 0dB for the Series SEV and paired comparison subjective testing placed it significantly in front (46% preference versus 27%). The Parallel SEV had clear channel output SNRSEG 0.9 dB lower than the Series/Parallel SEV at 9.1dB and paired comparison subjective testing placed it significantly behind (19% preference versus 42%). In a noisy channel with nominal BER of 0.5% the Parallel SEV had greater output SNRSEG by 0.8dB at 3.6dB than the Series/Parallel SEV, however this was not reflected in the paired comparison subjective testing which showed the Series/Parallel SEV had the subjectively superior output (49% preference versus 32%).

The Series/Parallel SEV was also compared to a reference CELP coder of equivalent bit rate. Output SNRSEG of both coders in clear channel was 10.0dB, however the Series/Parallel SEV was considerably more subjectively preferable (48% preference versus 21%). At a channel BER of 0.5%, output SNRSEG of the Series/Parallel SEV was 0.4dB greater at 2.7dB and their outputs were equally subjectively preferable (34% preference versus 32%). These tests were also performed with the Series/Parallel SEV utilising modified codebook adaption B. Its output SNRSEG in clear channel was reduced by 0.9dB to 9.1dB and it was also significantly less subjectively preferable than the reference celp coder (11% preference versus 61%). This situation was dramatically reversed at a nominal channel error rate of 0.5%. The Series/Parallel SEV

now had output SNRSEG 2.2dB greater than the reference CELP coder at 4.9dB and it was now considerably more subjectively preferable (47% versus 24%). This tradeoff of a small reduction in clear channel performance has significantly improved the Series/Parallel vocoders error robustness.

The results of this chapter have shown the excellent performance of the Series/Parallel Self Excited Vocoder. Performance is superior to CELP techniques and error robustness is comparable and can be enhanced further by utilising modified codebook adaptions. Further work is now required to develop a channel coder to further improve the error robustness of particularly sensitive bits, notably the codebook delay parameters. In addition, work is required to reduce the computational complexity of this speech coder to give a practical real time implementation.

Chapter 8

Conclusions

Chapter 8

Conclusions

The considerable growth in the use of cellular telephones over the past decade, coupled with future anticipated demands and the move to digital speech coding, has prompted much research into speech coding at rates below 8kbit/s. This thesis reports a three year study into the use of the Self-Excited Vocoder (SEV) for this application.

In the SEV, theoretically, the synthesis filter excitation is derived entirely from the previous history of the excitation. In practice, this is achieved using a conventional Linear Prediction Long Term Predictor (LTP), with no input. This is termed the Self-Exciting LTP (SE-LTP). It is demonstrated, in this thesis as by other researchers that, after initialisation, excellent speech quality can be achieved. However, such a simplistic scheme has very poor inherent error robustness and in a practical mobile telephony scheme, channel errors are generally unavoidable.

The perceptually weighted, analysis-by-synthesis SE-LTP was studied in detail. Initialisation techniques which combine fixed and adaptive portions of the adaptive codebook were used to enable error robust performance. The methods utilised were the Partial Fixed Codebook (PFC) and the Separate Fixed Codebook (SFC), they both had equivalent performance. A SEV with a single SE-LTP stage had superior speech quality than a CELP coder with a single fixed codebook stage of equivalent size. The SEVs output SNRSEG was 11% higher at 6.67dB and its synthetic speech output was eight times more subjectively preferable. However, the degradation of its speech quality in noisy channels as the percentage of channel errors increased was greater than that of the CELP coder and the latter had higher output SNRSEG above 0.25% BER. A conventional LTP was then incorporated into both CELP and SEV vocoders, giving them equal output SNRSEG of 10.0dB, there was no longer any advantage in using a SE-LTP stage rather than a CELP stage as the primary excitation source.

The adaption mechanism of the SE-LTP codebook was modified, making the adaption independent of the SE-LTP gain, this proving one of the most error sensitive parameters. Error robustness of the SEV was further improved. Output SNRSEG without the LTP at 1% BER was increased from 1.2dB to 2dB and with the LTP at 0.5% BER from 0dB to 2dB. In addition, a more perceptually pleasing characteristic distortion was produced in noisy channels. These improvements were at the expense of a small drop in clear channel performance (0.2dB of output SNRSEG for the single SE-LTP SEV).

Chapter 7 compared three different configurations of twin predictor SEVs. These were the Series and Parallel SEVs, which are already subjects in open literature, and the authors Series/Parallel SEV which combined features of both coders. In clear channel, objective performance of Series/Parallel and Series SEVs was equal (with output SNRSEG of 10.0dB) and the subjective quality was very close (32% versus 41% respectively). As the BER was increased, speech quality of the Series/Parallel degraded significantly more slowly. (At 0.5% BER, output SNRSEG was 2.8dB greater with subjective preference of 46% versus 27%). The Parallel SEV had inferior clear channel performance and although degradation in noisy channels followed a lower gradient, it remained significantly subjectively less preferable to the Series/Parallel SEV.

The Series/Parallel SEV was also compared to a reference CELP coder of equal bit rate. The clear channel objective performance of both coders was equal (with output SNRSEG of 10.0dB), however the Series/Parallel SEV was subjectively preferable (48% versus 21%). As channel errors were introduced, and as the BER increased, both vocoders degraded at an equal rate and subjective preference became equal. The tests were also performed with the Series/Parallel SEV utilising modified codebook adaption B. Its clear channel performance was significantly reduced (output SNRSEG was down 0.9dB to 9.1dB and it became clearly less subjectively preferable with 11% versus 61%). However as the BER increased, degradation was significantly reduced and the

SEV's speech quality became significantly superior to that of the CELP coder (at 0.5% BER output SNRSEG was 2.2dB greater and the SEV was significantly more subjectively preferable (47% versus 24%).

The results of this thesis have shown the excellent performance of the Series/Parallel Self Excited Vocoder. Performance is superior to CELP techniques and error robustness is comparable and can be enhanced further by utilising modified codebook adaptions. Further work could be to develop a channel coder to further improve the error robustness of particularly sensitive bits, notably the codebook delay parameters. In addition, further work could be to reduce the computational complexity of this speech coder to give a practical real time implementation.

Chapter 9

References

Chapter 9

References

1. Atal, B.S. and Schroeder, M.R. "Predictive Coding of Speech and Subjective Error Criteria." *IEEE Trans. on Acous., Speech, and Signal Proc.* Vol. ASSP-27, No. 3, (June 1979): 247-254.

2. Atal, B.S. "Predictive Coding of Speech at Low Bit Rates." *IEEE Trans. Commun.* Vol. COM-30, (April 1982): 600-614.

3. Atal, B.S. and Remde, J.R. "A New Model LPC Excitation Producing Natural Sounding Speech at Low Bit Rates." *Proc. Of 1982 IEEE Int. Conf. on Acous., Speech,* and Signal Proc. (May 1982): 614-617.

4. Atungsiri, S.A., Kondoz, A.M. and Evans, B.G. "Error Detection and Control for the Parametric Information in CELP Coders." *Proc. 1990 IEEE Int. Conf. on Acous. Speech and Signal Proc.* (May 1990).

5. Barnwell, T.P. III, Rose, R.C., and Mcgrath, S. "The Design and Performance of a Real-Time Self-Excited Vocoder." *Proc. Mobile Satellite Commun. Conf.* (May 1988).

6. Campbell, J.P., Welch, V.C. and Tremain, T.E. "An Expandable Error-Protected 4800 BPS CELP Coder (U.S. Federal Standard 4800 BPS Voice Coder)." *Proc. Of 1989* Int. Conf. on A*cous. Speech and Signal Proc.* 735-738

7. Chandra, S. and Lin, W.C. "Experimental Comparison Between Stationary and Nonstationary Formulations of Linear Prediction Applied to Voiced Speech Analysis." *IEEE Trans. on Acous. Speech. and Signal Proc.* Vol. ASSP-22, No. 6, (Dec 1974):

403-415.

8. Coleman, A.E., Gleiss, N. and Usai, P. "Quality Assessment of Speech Coders for Mobile Radio Application." *CSELT. Technical Reports* Vol. XVI No. 2 (March 1988).

9. Dankberg, M.D. and Wong, D.Y. "Development of a 4.8-9.6 kbit/s RELP Vocoder." *Proc of 1979 IEEE Int. Conf. on Acous., Speech, and Signal Proc.* 554-7

10. Dankberg, M.D., Lltis, R., Saxton, D. and Wilson, P. "Implementation of the RELP Vocoder using the TMS320" *Proc of 1984 IEEE Int. Conf. on Acous., Speech and Signal Proc.* (March 1984)

11. Dettmer, R. "GSM: European Cellular Goes Digital." *IEE Review* (Jul/Aug 1991): 253-257.

12. Federal Standard 1016: "Telecommunications: Analogue to Digital Conversion of Radio Voice by 4,800 bit/second Code Excited Linear Prediction (CELP)." *Second Draft* (Nov 1989).

13. Flanagan, J.L. *Speech Analysis Synthesis and Perception* Springer-Verlag, 1972.

14. Flanagan, J.L., Schroeder, M.R., Atal, B.S., Crochiere, R.E., Jayant, N.S. and Tribolet, J.M. "Speech Coding," *IEEE Trans Commun.,* Vol COM-27, No. 4, (April 1979): 710-737.

15. Galand, C., Rosso, M., Elie, Ph. and Lancon, E. "MPE/LTP Coder for Mobile Radio Application." *Speech Communication.* Vol. 7 No. 2 (July 1988).

16. Goodman, D.J., Goodman, J.S. and Chen, M. "Intelligibility and Ratings of Digitally Coded Speech." *IEEE Trans. on Acous. Speech and Signal Proc.* Vol. ASSP-26, No. 5, (Oct 1978): 403-409.

17. GSM Recommendation: 06.10 "GSM Full Rate Speech Trancoding."

18. Hanes, R.B. and Atkins, P. "The UK. Candidate 16kbit/s Codec for the GSM. Pan-European Study on Digital Cellular Land Mobile Radio." *Speech Communication.* Vol. 7 No. 2 (July 1988).

19. Holbeche, R.J., (Ed), *Land Mobile Radio Systems.* IEE Telecommunication Series, Vol. 14, Peter Peregrinus Ltd, 1985.

20. Hedelin, P. "Relp-Vocoding with Uniform and Non-uniform Down-Sampling" *Proc of 1983 Int. Conf. on Acous., Speech, and Signal Proc.* (April 1983): 1320-3

21. "IEEE Recommended Practice for Speech Quality Measurements." *IEEE Trans. Audio and Electroacoustics,* Vol. AU-17, No.3 (Sept 1969).

22. Kang, G.S. and Fransen, L.J. "Application of Line-Spectrum Pairs to Low-Bit-Rate Speech Coders." *Proc. of 1985 Int. Conf. on Acous. Speech and Signal Proc.* 244-247.

23. Kleijn, W.B., Krasinski, D.J. and Ketchum, R.H. "Improved Speech Quality and Efficient Vector Quantisation in SELP." *Proc 1988 IEEE Int. Conf. on Acous. Speech and Signal Proc.* (April 1988).

24. Kroon, P., Deprettere, E.F. and Sluyter, R.J. "Regular Pulse Excitation - A Novel Approach to Effective and Efficient Coding of Speech." *IEEE Trans. on Acous., Speech*

and Signal Proc. Vol. ASSP-34, No. 5, (Oct 1986): 1054-1063.

25. Kroon, P. and Atal, B.S. "Pitch Predictors with High Temporal Resolution." *Proc.* Of 1990 Int. Conf. on Acous. Speech and Signal Proc. 661-664

26. Kubota, H., Yokokura, A., Kikuchi, T. and Koyama, M. "High Capacity Automobile Telephone System -Part 1 System Outline." *Japan Telecommunications Review,* (Jan 1979): 44-53.

27. Kuwabara, H. and Ohgushi, K. "Role of Formant Frequencies and Bandwidths in Speaker Perception." *Electronics and Communications in Japan* Vol. 70, Part 1, No. 9, (Sept 1979): 11-21.

28. Lazzari, V., Montagna, R., Sereno, D. and Rusina, A. "Comparison of two Speech Codecs for DMR. Systems." *Speech Communication* Vol. 7 No. 2 (July 1988).

29. Lee, J.I. and Un, C.K. "Multistage Self-Excited Linear Predictive Speech Coder." *Electronics Letters* Vol 25 No. 18 (31st August 1989).

30. LeRoux, J. and Gueguen, C. "A Fixed Point Computation of Partial Correlation Coefficients." *IEEE Trans. Acous., Speech, and Signal Proc.,* Vol ASSP-25 (June 1977): 257-259.

31. Makhoul, J. and Berouti, M. "High-Frequency Regeneration in Speech Coding Systems." *Proc. Of 1979 IEEE Int. Conf. on Acous., Speech, and Signal Proc.* 428-431

32. Mattson, T. and Uddenfeldt, J. "Digital Versus Analogue Cellular Radio - Subjective Quality." *Second Seminar on Land Mobile Radio Communication,* Stockholm, (October

1986).

33. Menez, J., Galand, C., Rosso, M. and Bottau, F. "Adaptive Code Excited Linear Predictive Coder (ACELPC)." *Proc. 1989 IEEE Int. Conf. on Acous. Speech and Signal Proc.* (April 1989).

34. Menez, J., Galand, C. and Rosso, M. "A 2ms Delay Adaptive Code Excited Linear Predictive Coder." *Proc. 1990 IEEE Int. Conf. on Acous. Speech and Signal Proc.* (May 1990).

35. Natvig, J.E. "Evaluation of Six Medium Bit-Rate Coders for the Pan-European Digital Mobile Radio System." *IEEE Journal on Selected Areas of Communication.* Vol. 6 No.2 (Feb 1988).

36. Nieminen, J.U. "The Project Nordic Mobile Telephone." *Telecommunications (USA)* (April 1982): 52D-52H.

37. Papamichalis, P.E. *Practical Approaches to Speech Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

38. Perkins, J. "Minimum Shift Keying for Digital Cellular Radio." *MSc Thesis* University of Bath (1988).

39. Rabiner, L.R. "Digital Formant Synthesiser for Speech Synthesis." *J. Acoust. Soc. Am.* Vol 43 (1968): 822-828.

40. Ramachandran, R.P. and Kabal, P. "Pitch Prediction Filters in Speech Coding." *IEEE Trans. on Acous. Speech. and Signal Proc.* Vol. 37, No. 4 (April 1989): 467-478.

41. Ramstad, T.A. "Sub-Band Coder with a Simple Adaptive Bit Allocation Algorithm: A Possible Candidate for Digital Mobile Telephony?" *Proc. Of 1982 IEEE Int.* Conf. on Acous., Speech, and Signal Proc. (1982).

42. Rose, R.C. and Barnwell, T.P. "The Self-Excited Vocoder - An Alternative Approach to Toll Quality at 4800bps." *Proc 1986 IEEE Int. Conf. on Acous. Speech and Signal Proc.* (April 1986).

43. Rose, R.C. and Barnwell, T.P. "Design and Performance of an Analysis-by-Synthesis Class of Predictive Speech Coders." *IEEE Trans. on Acous. Speech and Signal Proc.* Vol. 38. No. 9, (Sept 1990): 1489-1503.

44. Saito, S. and Nakata, K. *Fundamentals of Speech Signal Processing.* Academic Press Inc. (London) Ltd, 1985.

45. Salami, R.A., Hanzo, L., and Appleby, D.G. "A Fully Vector Quantised Self-Excited Vocoder." *Proc. 1989 IEEE Int. Conf. on Acous. Speech and Signal Proc.* (April 1989).

46. Schafer, R.W. and Rabiner, L.R. "System for Automatic Formant Analysis of Voiced Speech." *J. Acous. Soc. Am.,* Vol. 47 (Feb 1970): 634-648.

47. Schroeder, M.R. "Models of Hearing." *Proc. Of IEEE,* Vol. 63, No. 9 (Sept 1975): 1332-1350.

48. Schroeder, M.R. and Atal, B.S. "Code-Excited Linear Prediction (CELP): High-Quality Speech at Very Low Bit Rates." *Proc of 1985 IEEE Int. Conf. on Acous., Speech, and Signal Proc.* 937-940.

49. Un, C.K. and Magill, D.T. "The Residual-Exited Linear Prediction Vocoder with Transmission Rate Below 9.6 kbit/s." *IEEE Transactions on Communications*. Vol. Com-23, No 12, (December 1975): 1466-1473

50. Tremain, T.E. "The Government Standard Linear Predictive Coding Algorithm: LPC-10." Speech Technology. (April 1982): 40-49.

51. Vary, P., Sluyter, R.J., Galand, C. and Rosso, M. "RPE-LPC Codec: The Candidate for the GSM Radio Communication System." *Int. Conf. on Digital Land Mobile Radio* Venice. (30 June - 3 July 1987): 507-516.

52. Vary, P., Hofmann, R., Hellwig, K. and Sluyter, R.J. "A Regular-Pulse excited Linear Predictive Codec." *Speech Communication*. Vol. 7 No.2 (July 1988).

53. Viswanathan, R. and Makhoul, J. "Quantisation Properties of Transmission Parameters in Linear Predictive Systems." *IEEE Trans. on Acous., Speech, and Signal Proc*. Vol. ASSP-23, No.3, (June 1975): 309-321.

54. Xydeas, C.S., Ireton, M.A. and Baghbadrani, D.K. "Theory and Real Time Implementation of a CELP Coder at 4.8kbits/second using Ternary Code Excitation." *Proc 5th Int. Conf. on Digital Proc. of Signals in Comms*. (Sept 1988).

55. Young, W.R. "Advanced Mobile Phone Service: Introduction, Background and Objectives." *Bell System Technical Journal*. Vol 58, No.1, (Jan 1979): 1-14.

56. Zinser, R.L. "An Efficient, Pitch Aligned High-Frequency Regeneration Technique for Relp Vocoders" *Proc of 1985 IEEE Int. Conf. on Acous., Speech, and Signal* Proc. 969-972.

Appendices

Appendix 1

Computational Aspects of Speech Coder Simulation

This thesis is concerned with the development of Self-Excited Vocoders and all studies take place in non real time simulations on general purpose computers. This appendix describes a suite of programs, modules and tools to assist this process. They are written in the high level "C" language. It is a powerful general purpose language that generates efficient, compact and portable code, it is also well suited to low level operations at bit or integer level. Borland International Turbo "C" professional 2.0 was available for personal computer use and this formed the editing/debugging environment. For tests on longer speech records, in the early stages of this research, a department Hewlett Packard 9000 series 800 minicomputer was used and, in the later stages, an Intel i860 "Number Smasher" PC coprocessor card was used. Software was written for portability, so once debugged in the PC environment, the same program could be compiled and run on the more powerful machine. This required software independent of machine specific constraints, such as word length.

Figure A1.1 illustrates the speech coder development process. The first stage is the algorithm design and its implementation in the high level language. The speech coder simulation is then run as an unquantised coder, where parameters are passed directly from encoder to decoder without any quantisation. This gives an initial indication of the speech coder performance, however this thesis is mainly concerned with the noisy channel performance which requires the testing of a fully quantised coder. A fully quantised coder transmits each parameter with a fixed number of bits. This is normally achieved by constructing quantisation tables for each parameter which consist of all available quantisation levels along with corresponding decision boundaries. The process of constructing the quantisation tables for a particular parameter starts by running the unquantised coder over a long and varied speech record and saving all the

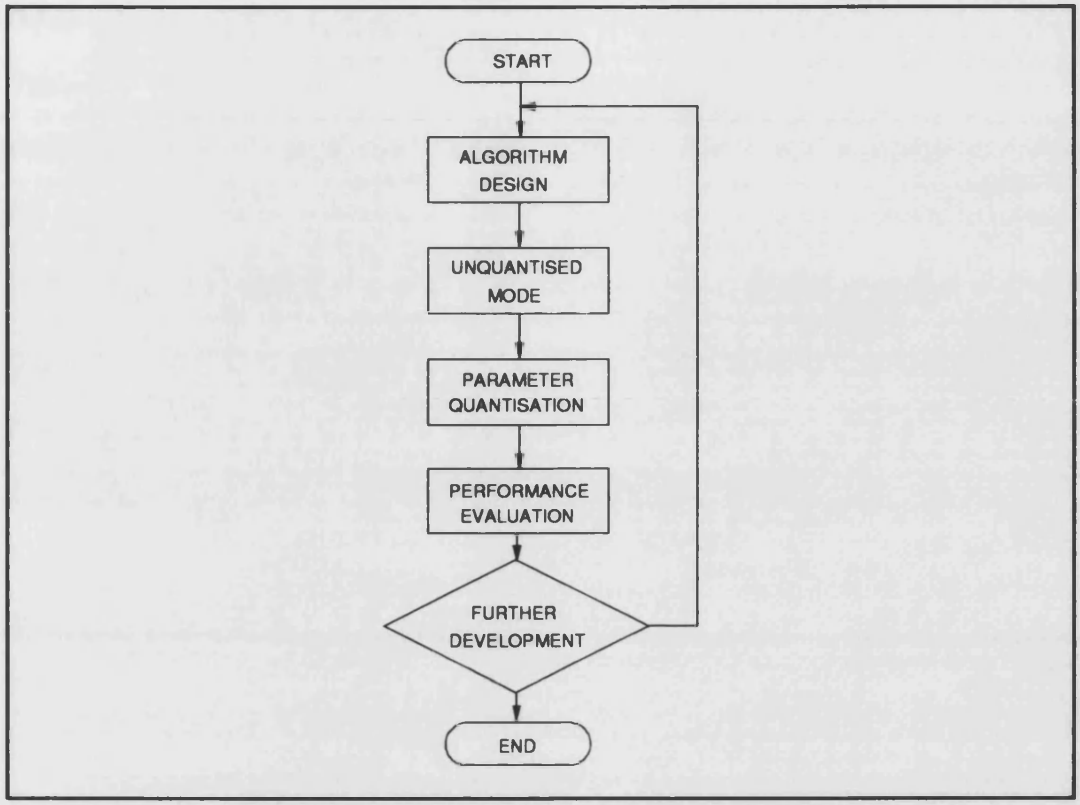

Figure A1.1 *The Speech Coder Development Process.*

calculated values to disk. Quantisation tables are constructed from these recorded values by a parameter quantiser program. When all the parameter quantisation tables have been determined, the speech coder could be run in quantised mode, giving a coded speech file from an input speech file. Noisy channel performance is evaluated by corruption of this coded speech file by a mobile radio channel simulator program. Synthetic speech is decoded from the corrupted code file by the speech decoder.

Section A1.1 describes the utilities for recording and replaying speech records. Section A1.2 describes the encoder/decoder module which formed the basis of all the analysis-by-synthesis speech coders of this thesis. Section A1.3 describes the gain parameter quantisation program. Finally, section A1.4 describes a versatile mobile radio channel simulator for introducing random channel errors into encoder code files before they are input to the decoder.

A1.1 Speech Record/Replay Utilities

The non-real time study of speech coders required the development of utilities for the recording and replaying of speech samples which operated in conjunction with IBM PC compatible housed DSP hardware. In addition software packages were available for displaying and editing waveforms and for performing signal processing operations.

Both a Loughborough Sound Images TMS 320C25 development board [A1.1] and an Ariel DSP32C development board [A1.2] were available. Hardware and Software were developed for replaying and recording speech to and from the PC hard disk using the TMS 320C25 board. The "Hypersignal" software [A1.3] package could be used for recording and replaying speech in conjunction with either of the two DSP boards. This could also show waveforms on the PC screen, allow zooming in and replaying of particular segments and also editing of speech recordings. Along with many other capabilities, Hypersignal was also able to construct and display spectrograms of speech waveforms.

A speech database was made from recordings of 32 speakers, 16 male and 16 female, each saying 20 sentences from two lists of the Harvard list of phonetically balanced sentences [A1.4]. Use of a recording studio, high quality equipment and considerable care gave high quality recordings. They were digitised using the Ariel DSP32C development board in conjunction with Hypersignal software. From all these speakers, 5 different speech test records were constructed by electronic editing. Each consisted of all the sentences in order, but spoken by 20 different speakers, with equal numbers of males and females. The sentences used, along with the breakdown of each test record, are given in appendix 2.

A1.2 Encoder and Decoder Implementation

The studies of chapters 4 and 5 identified the need for a general purpose speech coder module upon which all subsequent speech coders could be based. This module would

handle all the common functions of the experimental Self-Excited and CELP coders. Parameter optimisation would be by analysis-by-synthesis methods. Two modules were developed, firstly a combined encoder/decoder module allowing the easy passing of unquantised values from encoder to decoder, and secondly, a decoder module.

The initial requirement of the speech coder module was to provide a universal program "front end", which would handle all the disk I/O operations, enable setting up of the speech coder by command line switches, monitor coder throughput, and provide a means of loading initialisation sequences and quantisation tables.

The determination of synthesis filter parameters is common to all the speech coders studied and was incorporated within the general purpose coder module. Filter order could be either 8 or 12 and facility existed for using unquantised filter parameters. Synthesis filter parameters were determined over non-overlapping rectangular windowed frames of 160 samples. The perceptual error weighting sections were also part of the module along with switchable preemphasis and deemphasis functions.

Incorporation of excitation generators was designed to be as flexible as possible. They operated over subframes of 40 samples and as many stages could be incorporated as desired. These were essentially what the coder module lacked from being a complete speech coder. Facility existed for the passage of command line setup information to the excitation generator modules.

The decoder only module was entirely cut down from the encoder/decoder module and operated only in quantised mode.

The precise method for the error minimisation and the perceptual error weighting requires further explanation. Figure A1.2 shows the block diagram of the reference CELP coder in greater detail (previously shown as figure 6.1). Each candidate excitation sequence is convolved with the synthesis filter impulse response and this result is

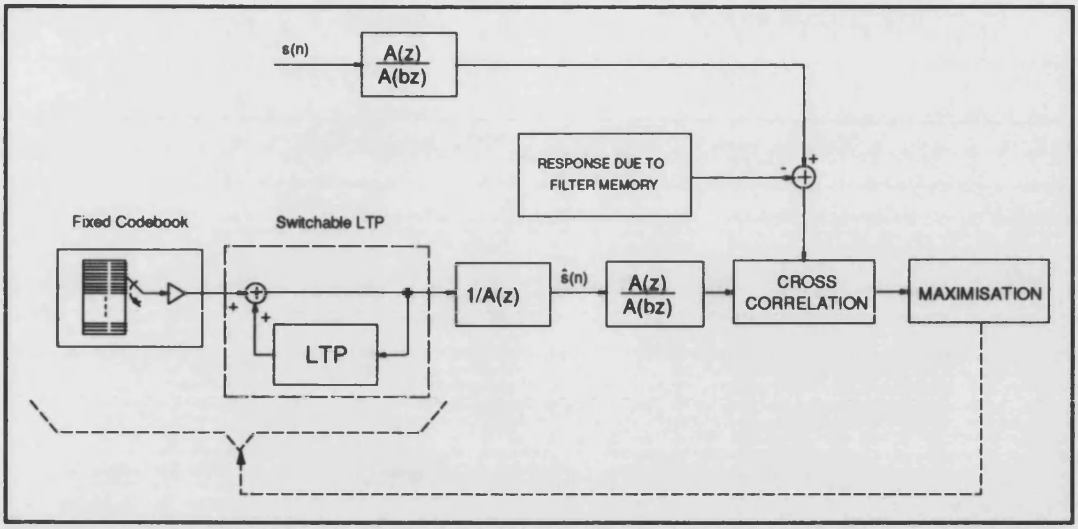

Figure A1.2 *Typical Speech Coder Block Diagram, By Way of Example, The Reference CELP Coder.*

cross-correlated with the target vector. The optimum excitation vector gives the highest result, this corresponds to the maximisation of $E'(d)$ in equation 3.64. Thus the optimum candidate excitation vector can be determined without calculation of any gain terms. Because of this method of error minimisation, perceptual error weighting is cannot be performed by filtering the difference signal $s(n) - \hat{s}(n)$. Instead, as shown in the block diagram of figure A1.2, input speech $s(n)$ is perceptually filtered by error weighting filter $A(z)/A(bz)$ before it is cross-correlated with synthetic speech $\hat{s}(n)$ which is also perceptually filtered. The synthetic speech is now passed through two filters, firstly synthesis filter $1/A(z)$ and secondly perceptual filter $A(z)/A(bz)$. It should be noted that dequantised parameters are used for the synthesis filter whereas unquantised parameters are used for the perceptual filter, otherwise both filters would be equivalent to single filter $1/A(bz)$.

The flowchart for the general purpose linear predictive speech encoder/decoder module is shown in figure A1.3, the first operation box marked "initialisation operations" includes setting the vocoder options such as the speech file to be processed, whether preemphasis and deemphasis are required, whether error weighting is required, whether the vocoder should operate in quantised or unquantised mode and whether any intermediate parameters, such as gains and delays should be output to separate files.

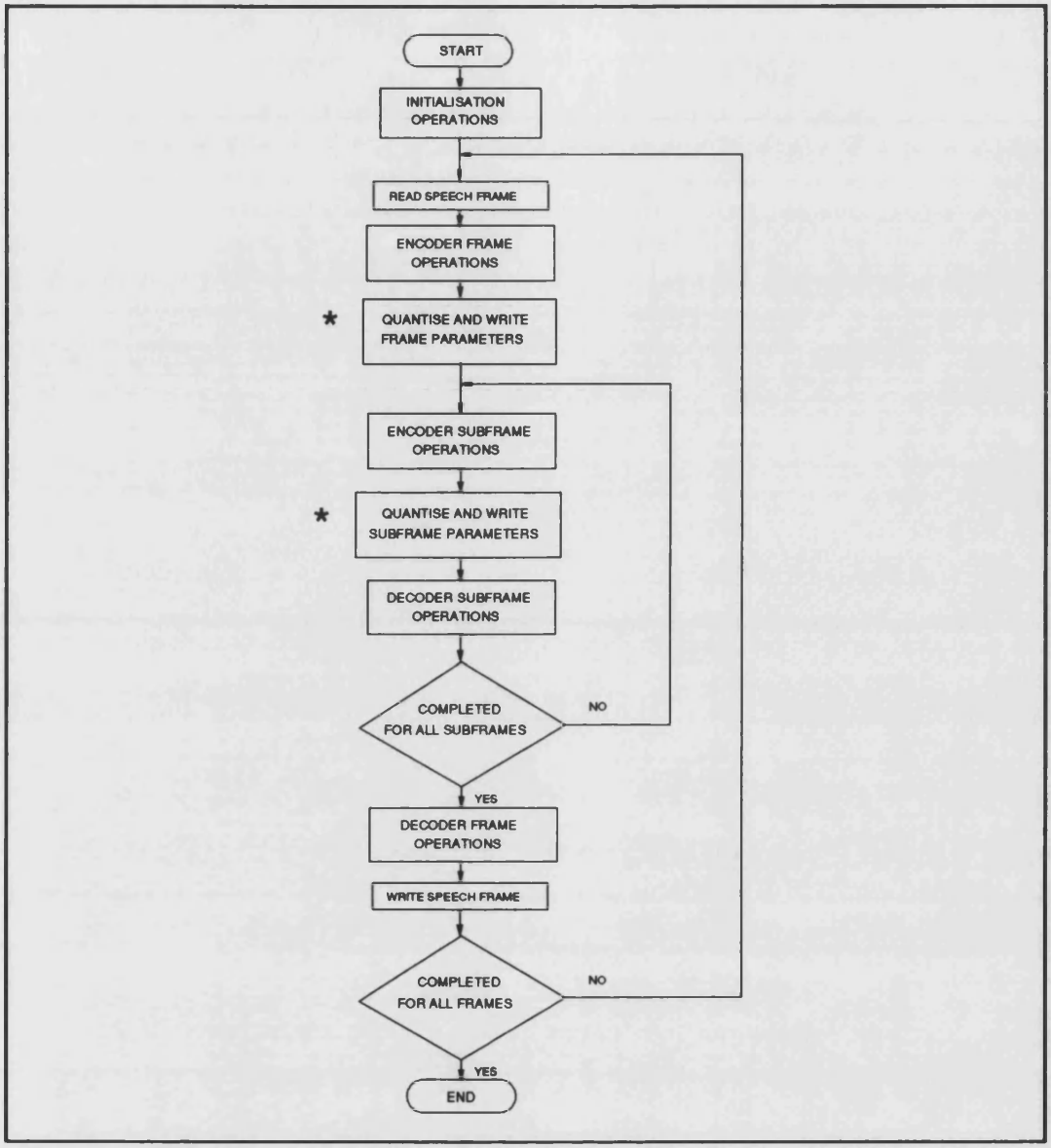

Figure A1.3 *Flowchart of Speech Coder Module, Operation Marked "*" Omitted in Unquantised Mode.*

This would also include setting of any switches within the subframe operations. Switches are normally set from the program command line, otherwise defaults would be used. If no filename is entered, the user will be prompted to supply one. The general encoder module assumes an input speech file to have the extension *.inp*, outputs a code file of the same name with extension *.cod* and outputs a speech file of extension *.out*. This is compatible with the authors speech record and playback facilities.

The next operations are the opening of input and output files, and the loading of any required initialisation sequences and quantisation tables. Processing now starts on speech frames. Framewise operations are performed on the first frame, such as preemphasis, LPC analysis and input speech perceptual weighting etc. In quantised mode, parameters are now quantised and output to the code file, this stage is omitted in unquantised mode.

Processing is now divided into individual subframes, carrying out the excitation generator functions which give the speech coder its personality such as self-excited analysis, code excited analysis etc. In quantised mode, parameters would be quantised and written to the code file.

Decoder subframe operations now take place where the synthesis filter excitation is reconstructed. The encoder and decoder operations are completed for all the subframes of the current speech frame. The final stage is the decoder frame operations which are short term synthesis filtering and deemphasis filtering. The entire process is repeated for all the speech frames of the input speech file.

Input and output speech was stored as 16 bit integers. The code file parameters were again stored as 16 bit integers using as many LSBs as were required (right justified). It is important, for channel simulation purposes, that all parameters are adjusted so they are always transmitted as positive integers, rather than as 16 bit twos complements when the actual parameter is negative. This allows the transmitted bits to be identified within the code file, allowing easy noisy channel simulation.

The flowchart for the general purpose speech decoder module is shown in figure A1.4. This is entirely "cut-down" from the general purpose encoder module just described. All the encoder steps are removed, and the decoder input is the code file and the output is speech. The decoder can only operate in quantised mode. The next sections describe frame and subframe operations in greater detail.

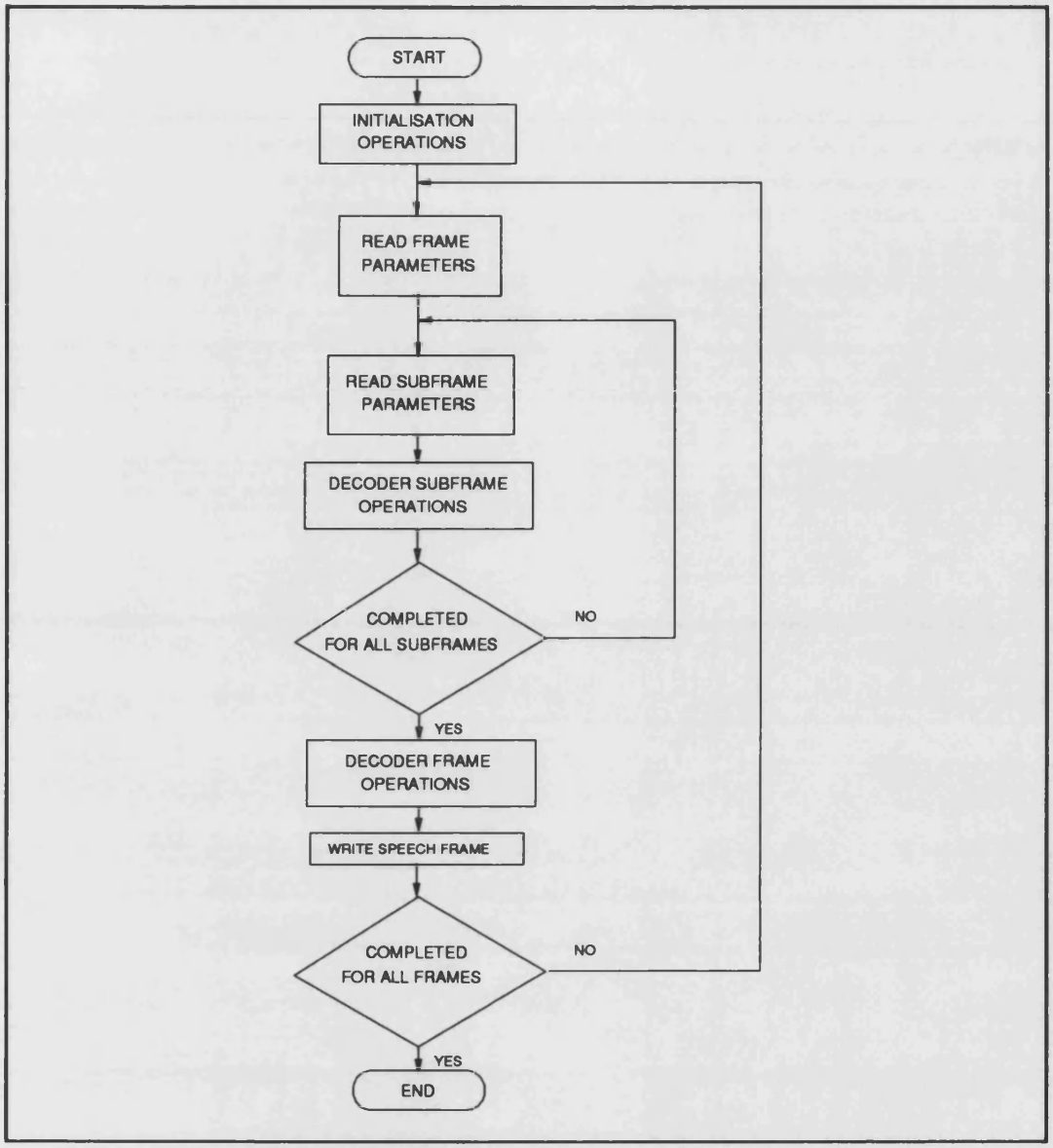

Figure A1.4 *Flowchart for Speech Decoder Module.*

1.2.1 Frame Operations

Figure A1.5 shows a flowchart of the frame operations in more detail. The first process is preemphasis of the input speech frame if required, followed by the determination of synthesis filter parameters. This is handled by routine Lpc_Anal which takes 160 speech samples and determines either 8 or 12 synthesis filter reflection coefficients. This process is performed by Schur recursion as described by [A1.5].

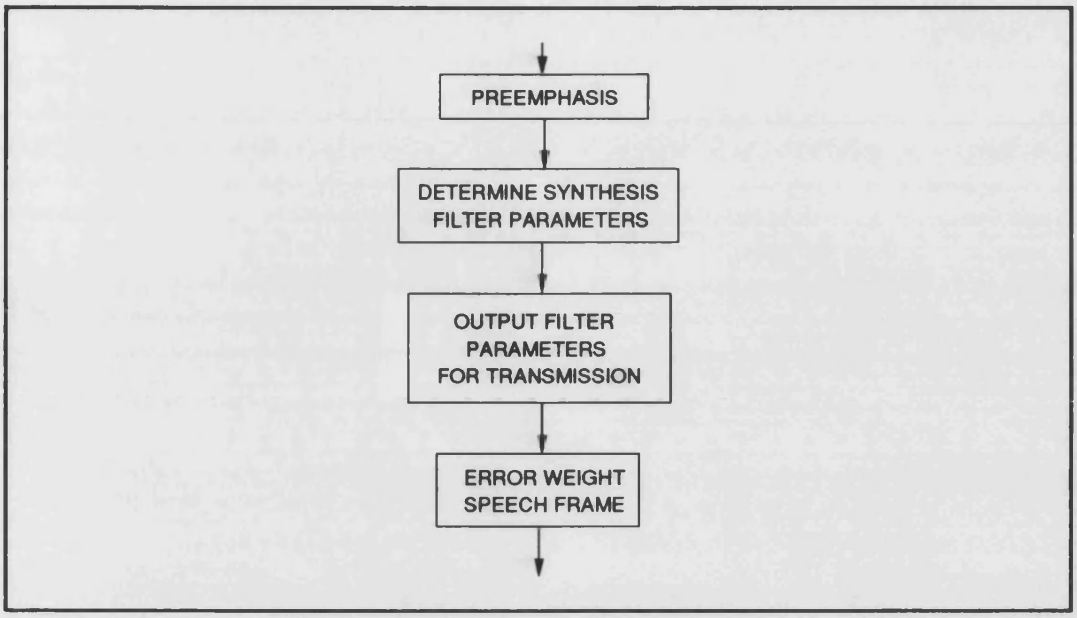

Figure A1.5 *Flowchart for Frame Operations.*

The reflection coefficients which have range $-1 \leq r_i \leq +1$ and are converted into Log-Area-Ratios due to their favourable quantisation characteristics [A1.6]. This conversion is strictly defined as

$$LAR_i = \log_{10}\left(\frac{1+r_i}{1-r_i}\right) \qquad A1.1$$

Since it is the companding characteristic that is important, the following segmented approximation is used

$$LAR_i = \begin{cases} r_i & |r_i| < 0.675 \\ sign[r_i] \times [2|r_i| - 0.675] & 0.675 \leq |r_i| < 0.950 \\ sign[r_i] \times [8|r_i| - 0.950] & 0.950 \leq |r_i| \leq 1.000 \end{cases} \qquad A1.2$$

This reduces highly computationally complex division and logarithm operations to much more simple add, multiply and compare. The inverse of this transformation is given by

$$f_i = \begin{cases} \hat{LAR}_i & |\hat{LAR}_i| < 0.675 \\ sign[\hat{LAR}_i] \times [0.500|\hat{LAR}_i| + 0.337500] & 0.675 \le |\hat{LAR}_i| < 1.225 \\ sign[\hat{LAR}_i] \times [0.125|\hat{LAR}_i| + 0.796875] & 1.225 \le |\hat{LAR}_i| \le 1.625 \end{cases} \qquad A1.3$$

The log-area-ratios have different dynamic ranges and different asymmetric distributions and are quantised with numbers of bits as given in tables A1.1 and A1.2, for 8th and 12th order filters respectively.

LAR No. i	Bits/LAR
1,2	6
3,4	5
5,6	4
7,8	3

Table A1.1 *Bit Assignment for Quantisation of LAR Coefficients, for 8th Order Filter.*

LAR No. i	Bits/LAR
1,2	6
3,4	5
5,6,7,8,9,10	4
11,12	3

Table A1.2 *Bit Assignment for Quantisation of LAR Coefficients, for 12th Order Filter.*

Each Log-Area Ratio is encoded by

$$LARc[i] = (A[i]*LAR[i]) + B[i] - MIC[i] \qquad A1.4$$

And decoded by

$$LAR[i] = \frac{LARc[i] + MIC[i] - B[i]}{A[i]} \qquad A1.5$$

Values of Ai, Bi and $MICi$ are given in tables A1.3 and A1.4 for 8th and 12th order synthesis filters respectively, along with the dynamic range over which values are quantised.

LAR	Range	A	B	MIC
1	-1.6,1.5	20	0	-32
2	-1.6,1.5	20	0	-32
3	-1.0,0.55	20	4	-16
4	-0.55,1.0	20	-5	-16
5	-0.6,0.5	13.637	0.184	-8
6	-0.3,0.7	15	-3.5	-8
7	-0.4,0.44	8.334	-0.666	-4
8	-0.2,0.6	8.824	-2.235	-4

Table A1.3 *Log-Area Ratio Range and Quantisation Information for 8th Order Synthesis Filter.*

LAR	Range	A	B	MIC
1	-1.6,1.5	20	0	-32
2	-1.6,1.5	20	0	-32
3	-1.0,0.55	20	4	-16
4	-0.55,1.0	20	-5	-16
5	-0.7,0.8	10	-1	-8
6	-0.7,0.8	10	-1	-8
7	-0.7,0.8	10	-1	-8
8	-0.7,0.8	10	-1	-8
9	-0.7,0.8	10	-1	-8
10	-0.7,0.8	10	-1	-8
11	-0.3,0.4	10	-1	-4
12	-0.3,0.4	10	-1	-4

Table A1.4 *Log-Area Ratio Range and Quantisation Information for 12th Order Synthesis Filter.*

The next stage is to convert synthesis filter reflection coefficients into filter parameters, enabling calculation of error weighting filter parameters. This is achieved using the equations 3.31a and 3.31b. They are then converted back to reflection coefficients by equations 3.32a and 3.32b.

To avoid transients arising from the abrupt changing of filter coefficients, linear interpolation is applied in the transition between two successive parameter sets. This occurs over the first 5ms of each frame and is performed on both synthesis filter coefficients and perceptual filter coefficients..

1.2.2 Subframe Operations

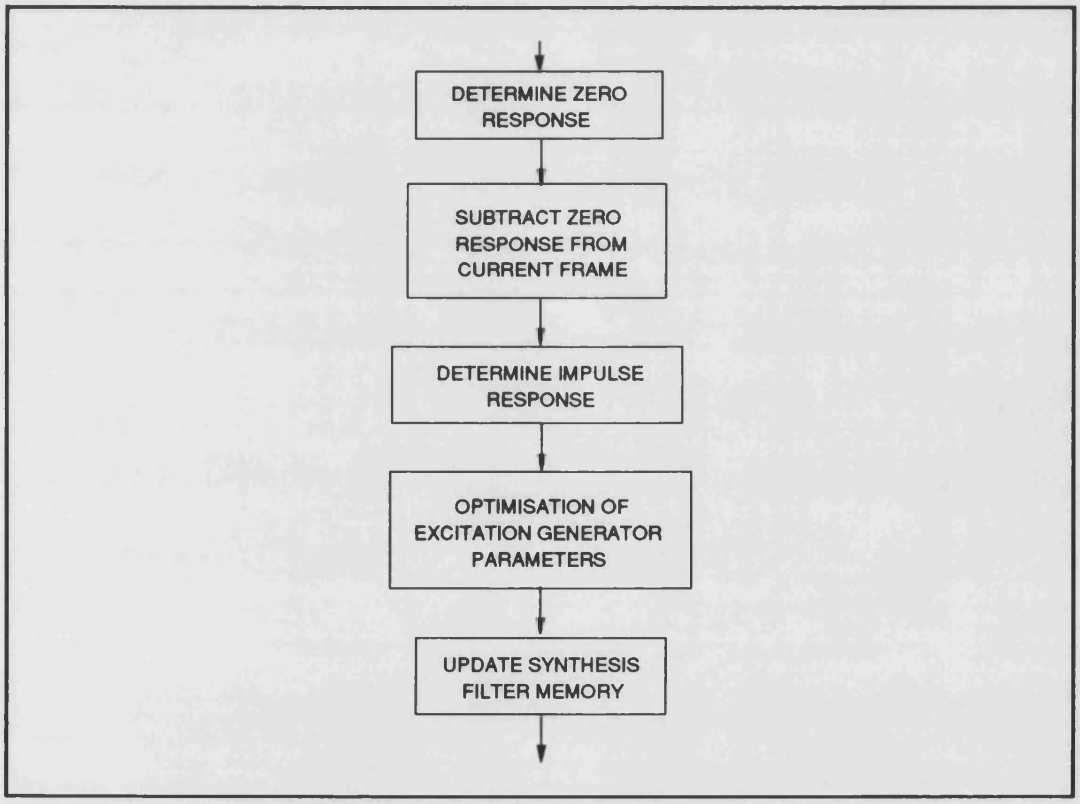

Figure A1.6 *Flowchart for Subframe Operations.*

The general subframe operations, which are is part of the general speech coder module, are the calculation of the target speech vector and synthesis filter impulse response. These are the first three and final stages of the subframe operations flowchart of figure A1.6. They utilises subroutine Enc_Synt which takes a 40 sample vector uses it as input

to the synthesis filter $1/A(z)$. As this routine was designed for the speech encoder, it has facility for inclusion of the perceptual error weighting filter $A(z)/A(bz)$. The flowchart for subroutine Enc_Synt is shown in figure A1.7. The subroutine operates in three different modes, dependent upon the value of variable *FILMEM* upon entry, which controls the maintaining of the filter memory.

Firstly, the zero response of the synthesis filter is determined with the filter memory remaining from the synthesis of the previous subframe. The input vector to Enc_Synt is the zero vector, and the value of *FILMEM* is set to 1, which makes the routine set the filter memory equal to the saved values (including both synthesis filter and perceptual filter). The zero vector is then input to synthesis filter and if the error weighting is enabled, the result is input to perceptual filter. On exit from the routine, the values of memory contents are not saved. The zero response is then subtracted from the input speech vector (after perceptual filtering if enabled) to give the target vector for optimisation of excitation generator parameters. The next stage is to determine the synthesis and perceptual filter impulse response. This also uses routine Enc_Synt, however variable *FILMEM* is set to zero, which sets filter memory to zero on entry and again does not save its contents at exit. The next stage is the optimisation of excitation generator parameters which will be described later and the final stage is the updating of the synthesis filter memory. This is performed after the synthetic excitation signal has been determined and this signal is input to routine Enc_Synt. Value of *FILMEM* is set to 2 which sets filter memory equal to saved values on entry and saves the final filter memory contents at exit. One final note about routine Enc_Synt is the apparent filtering by $1/A(z)$ followed by inverse filtering $A(z)$ which would appear to give the identity operation. The former operation uses dequantised filter parameters whereas the latter operation uses unquantised filter parameters.

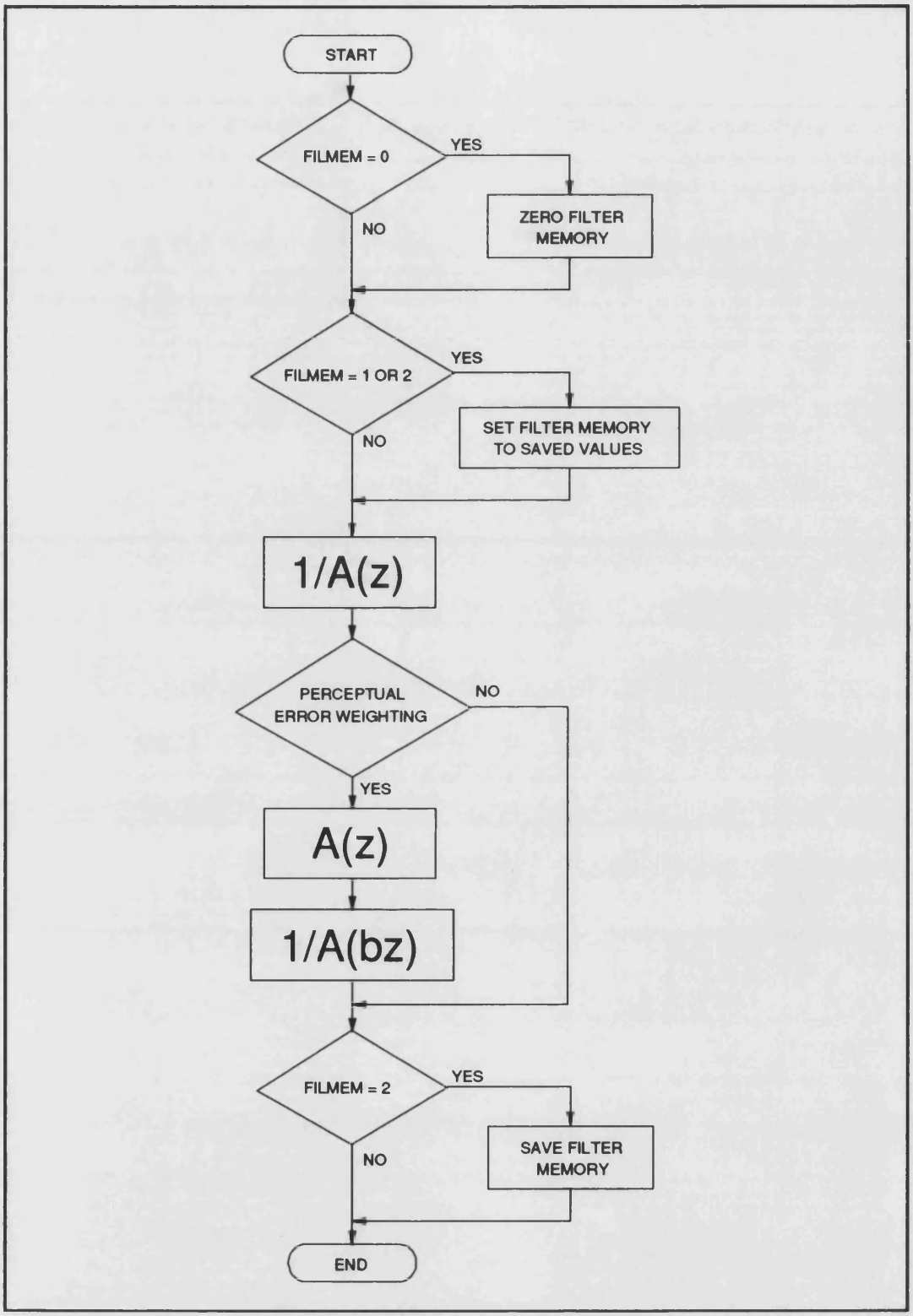

Figure A1.7 *Flowchart for Subroutine Enc_Synt.*

The excitation generator stages govern the nature of the speech coder. CELP coders have been studied, with and without LTPs, SEVs have been studied, including single predictor, Series, Parallel and Series/Parallel variants. This personality is achieved within the excitation generator stages. The all operate in the same manner, candidate excitations are convolved with the synthesis filter impulse response and the result cross-correlated with the target vector to find the optimum excitation. Where multiple excitation generators are used, a new target vector is calculated after the first stage of optimimsation. The precise nature of the excitation generators has been described within the text.

1.2.3 Implementation of a Non-Integer Pitch LTP

The reference CELP coders and the Series SEV all used the same LTP code section. It had an adaptive codebook of 147 elements giving 128 integer delays, and 128 non-integer delays. The precise allocation of delays and corresponding codebook index is listed in appendix 3, this allowed modelling of delays from 2.5 to 18.5ms. Operation of the LTP has been adequately covered in chapter 3, this section will expand upon the non-integer pitch aspect.

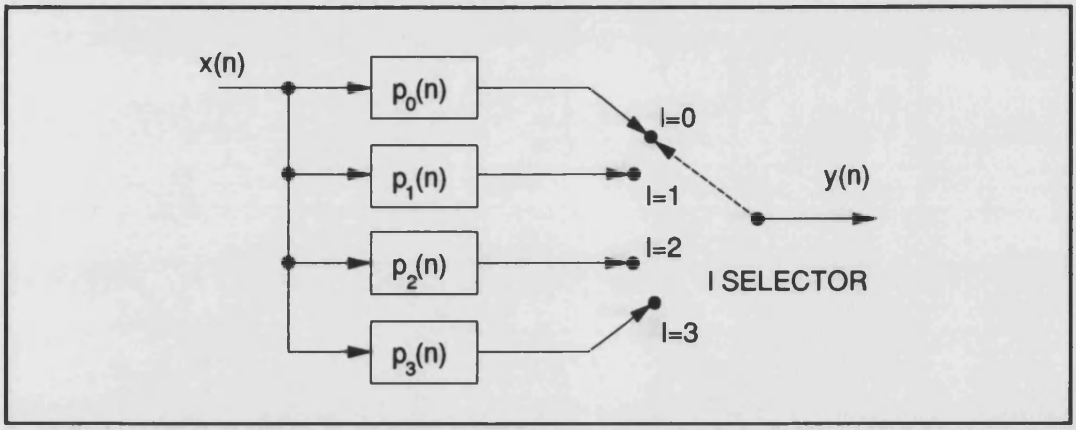

Figure A1.8 *A Polyphase Network Implementation of a Quarter Sample Delay Network.*

Non-integer pitch interpolation was performed by two subroutines, one determined quarter sample delays and the other third sample delays. Chapter 3 described the up-sampling and low pass filtering operation. This can be more efficiently implemented

using a polyphase structure [A1.7] as illustrated in figure A1.8. This structure realises the sampling rate increase and the low pass filtering operation. A delay of l/D samples at the original sampling rate corresponds to each position of the fixed arm commutator, where D corresponds to the sampling rate increase factor. The polyphase filters are obtained from the coefficients of the low-pass filter $h(n)$, $n = 0, 1, \ldots, N-1$ according to

$$p_l(k) = h(kD - l) \qquad 0 \le l \le D-1, \quad k = 0, 1, \ldots, q-1 \qquad A1.6$$

where $h(n) = 0$ for $n < 0$. The number of coefficients of the polyphase filter p_l is given by

$$q = [N/D] \qquad A1.7$$

where [ ] denotes rounding to the next largest integer. For each value of the delay l/D the corresponding l-th polyphase filter branch output is given by

$$y(n) = \sum_{k=0}^{q-1} p_l(k)x(n-k) \qquad A1.8$$

Taking into account the delay I of the low-pass filter, the expression for the pitch predictor with an effective non-integer delay $d + l/D$ becomes

$$P_e(z) = \gamma \sum_{k=0}^{q-1} p_l(k) z^{-(d-I+k)} \qquad A1.9$$

where γ represents the pitch predictor coefficient.

The low-pass filter used was a $\sin(x)/x$ function weighted with a hamming function. This approximates an ideal low-pass filter with passband between 0-4000 Hz. For each value of D, the length of the filter was chosen such that the delay I at the lower sampling frequency was equal to 16 samples.

A1.3 Parameter Quantisation

LTP and SE-LTP gain parameters are quantised using maximum entropy quantisation [A1.8], where the probability of a value being between any two consecutive decision

levels is the same. The corresponding quantisation level between two adjacent decision levels is set at the centroid of the area. The quantisation levels have equal probability of occurrence.

A Parameter Quantiser program has been designed to determine the discrete decision levels and the corresponding quantisation levels for a particular parameter. The program requires as input, a long record of unquantised values obtained by running a speech coder over a long and varied speech record. During this research, a 53 second record consisting of 20 sentences from the Harvard list of phonetically balanced sentences spoken by 10 male and 10 female speakers was used. The quantiser program constructed the two lists of values, referred to as a quantisation table, which are stored as disk files and are input into speech coders during their initialisation stage. The program works in two different ways, depending upon whether a "zero threshold" is required, any absolute value less than the zero threshold is quantised to zero. This prevents quiet spurious outputs when the speech encoder input is silent.

The quantisation program flowchart is depicted in figure A1.9. The input to this program is a list of ascii text values of any magnitude. The initialisation operations include setting the number of quantisation levels, opening the input file, reading its contents into memory, and the setting of a zero threshold if required. If a zero threshold is entered, then the first decision level $D0$ is set equal to it. The next stage is to calculate the number of values that each interval will contain. If no zero threshold is required, this is simply the number of values in the input parameter file divided by the number of quantisation levels. Alternatively with a zero threshold, this is the number of values of absolute value greater than the threshold, divided by the number of quantisation levels minus one. With the number of values in each interval known, the determination of decision levels can begin.

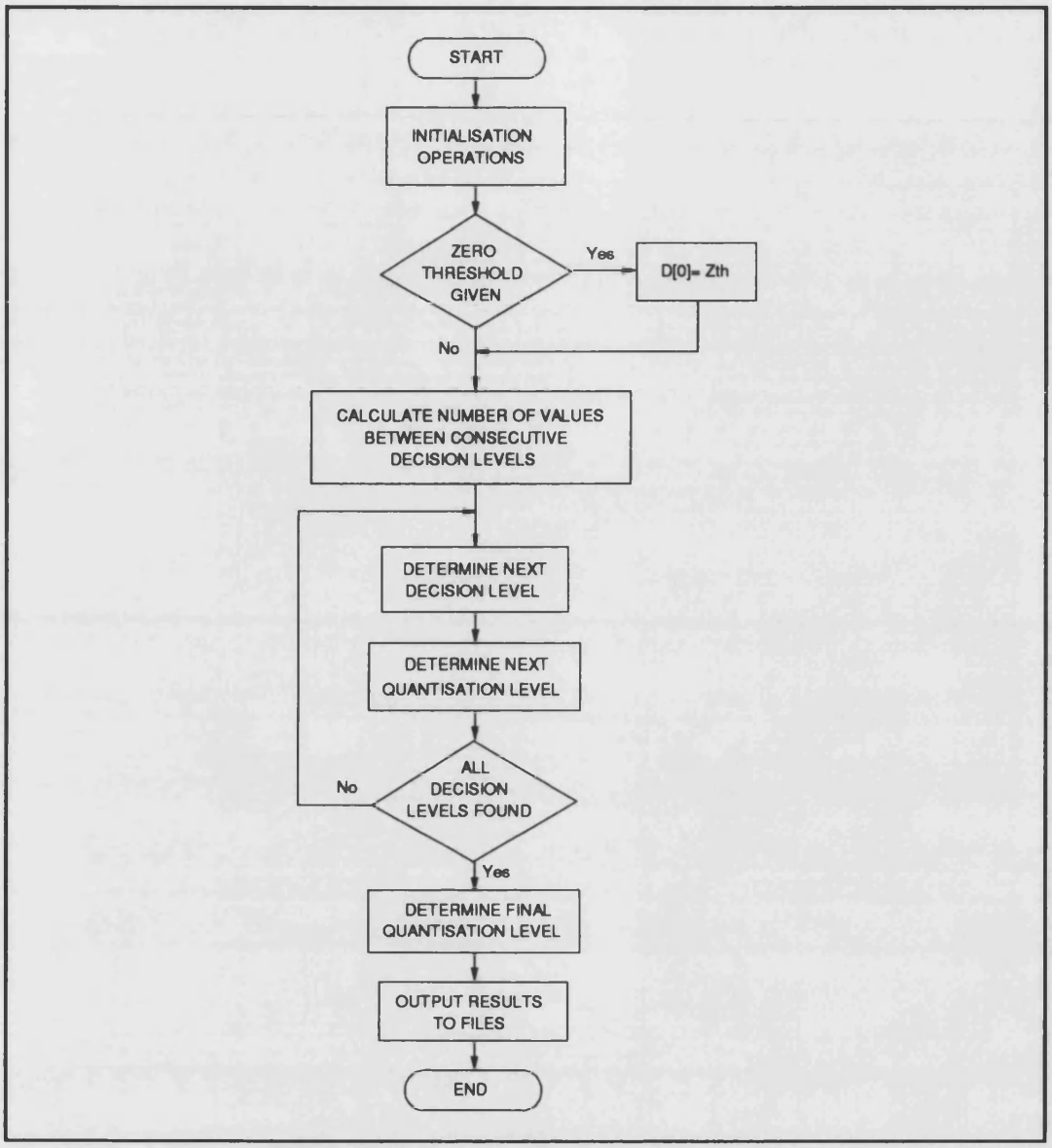

Figure A1.9 *Flowchart of Parameter Quantiser.*

The way the next decision level is found, is by starting with a working level, if the number of values between the previous decision level and this working level is less than the required number in the interval, the working level is increased. Alternatively if the number of values is greater, the working level is decreased. For every new working level, the list of input values is cycled through to determine the number of values now contained. Initially the working level is increased or decreased by 5%, when the

optimum decision level has been crossed, this increase or decrease is reduced to 0.5% and a nearer value found. Finally the change is set to 0.05% before giving a final value for the decision level.

With two successive decision levels known, the quantisation level for that interval can be determined by finding the mid range value of all parameter values within this interval. In all but the first and last quantisation level this gave a result very nearly equal to the mean value within the interval. With the first and last quantisation levels, the level was not influenced by exceptionally high or low parameter values as the mean value would be.

These two steps are repeated until all the decision levels have been found, in each case the initial working decision level started equal to the previously determined one. When all the decision levels have been found, the extreme quantisation levels are the mid range values of all parameter outside the extreme decision levels. The final stage of the parameter quantiser is to output result files of quantisation and decision levels.

A1.4 The Mobile Radio Channel Simulator

The mobile radio channel simulator was designed as a versatile tool to test speech coder operation over noisy mobile radio channels by introducing random bit errors into code files. All the non-real time implementations of speech coders, process speech and give an output code file. Each code file parameter is stored using the LSBs of a 16bit word. Thus a code file has many unused bits saved alongside the useful information. To randomly corrupt the information carrying bits, requires them to be identified from the unused ones. Extra versatility is obtained by allowing the corruption of different bits with different error probabilities.

The channel simulator is programmed via a channel simulation table (CST). A different one is written for each coder to be tested. This enables transmitted bits firstly to be

```
3.09023 0 0 0 0 0 0 0 0
;
; CST for Series/Parallel Self-Excited Vocoder
;
; filter parameters
:6  111111      LARc[1]
:6  111111      LARc[2]
:5  11111       LARc[3]
:5  11111       LARc[4]
:4  1111        LARc[5]
:4  1111        LARc[6]
:4  1111        LARc[7]
:4  1111        LARc[8]
:4  1111        LARc[9]
:4  1111        LARc[10]
:3  111         LARc[11]
:3  111         LARc[12]
; excitation parameters
; subframe 1
!8  11111111    Nc1(1) SE-LTP1 delay
!6  111111      bc1(1) SE-LTP1 gain
!8  11111111    Nc2(1) SE-LTP2 delay
!6  111111      bc2(1) SE-LTP2 gain
; subframe 2
!8  11111111    Nc1(2) SE-LTP1 delay
!6  111111      bc1(2) SE-LTP1 gain
!8  11111111    Nc2(2) SE-LTP2 delay
!6  111111      bc2(2) SE-LTP2 gain
; subframe 3
!8  11111111    Nc1(3) SE-LTP1 delay
!6  111111      bc1(3) SE-LTP1 gain
!8  11111111    Nc2(3) SE-LTP2 delay
!6  111111      bc2(3) SE-LTP2 gain
; subframe 4
!8  11111111    Nc1(4) SE-LTP1 delay
!6  111111      bc1(4) SE-LTP1 gain
!8  11111111    Nc2(4) SE-LTP2 delay
!6  111111      bc2(4) SE-LTP2 gain
;
; total 164 bits/frame (8.2Kbit/s)
;
```

Figure A1.10 *A typical Channel Simulation Table.*

recognized from an output code file and secondly, each bit in the frame to be errored with one of nine different error probabilities. A typical CST is shown in figure A1.10, this one being for the Series/Parallel SEV introduced in chapter 7.

The first line of the table consists of nine values. These are the normal probability functions of the error probabilities for the nine different probability indexes. The rest of the table divides into 3 fields.

The first field contains two entries. The first character is either ";", ":" or "!". The ";" indicates that the remainder of the line is a comment. The ":" indicates that the particular parameter is uncorrupted for every frame of the code file, regardless of any information on the remainder of the line. Whereas "!" indicates that the parameter bits should be subject to error as described by the remainder of the line. The second character is a count of the number of bits used in the code file word.

The second field indicates the probability index of each of these parameter bits. These are digits between 0 and 9. Digits 1 to 9 correspond to the normal probability functions given in the CST top line. Whereas 0 indicates that the particular bit is uncorrupted. In this table, the probability index of each bit in each frame is set to 1. The normal probability function corresponding to this index 1 is given in the top CST line as 3.09023, this corresponds to a BER of 0.1%. The rightmost digit corresponds to the LSB.

The third field is purely for comments, enabling easy understanding and reprogramming of the channel simulator.

The flowchart for the Channel Simulator is shown in figure A1.11, the first operation is the reading of the CST table, and setting up a structure containing all the essential data. Another important operation is to determine how many parameters each frame contains, this is equal to the number of non-comment lines in the CST.

The program proceeds and the first word is read from the input code file. If the CST line corresponding to that parameter started with "!", the main program route is followed and this parameter could be errored. A normal probability deviate is determined using utility routines from [A1.9]. This is compared with the normal probability function for

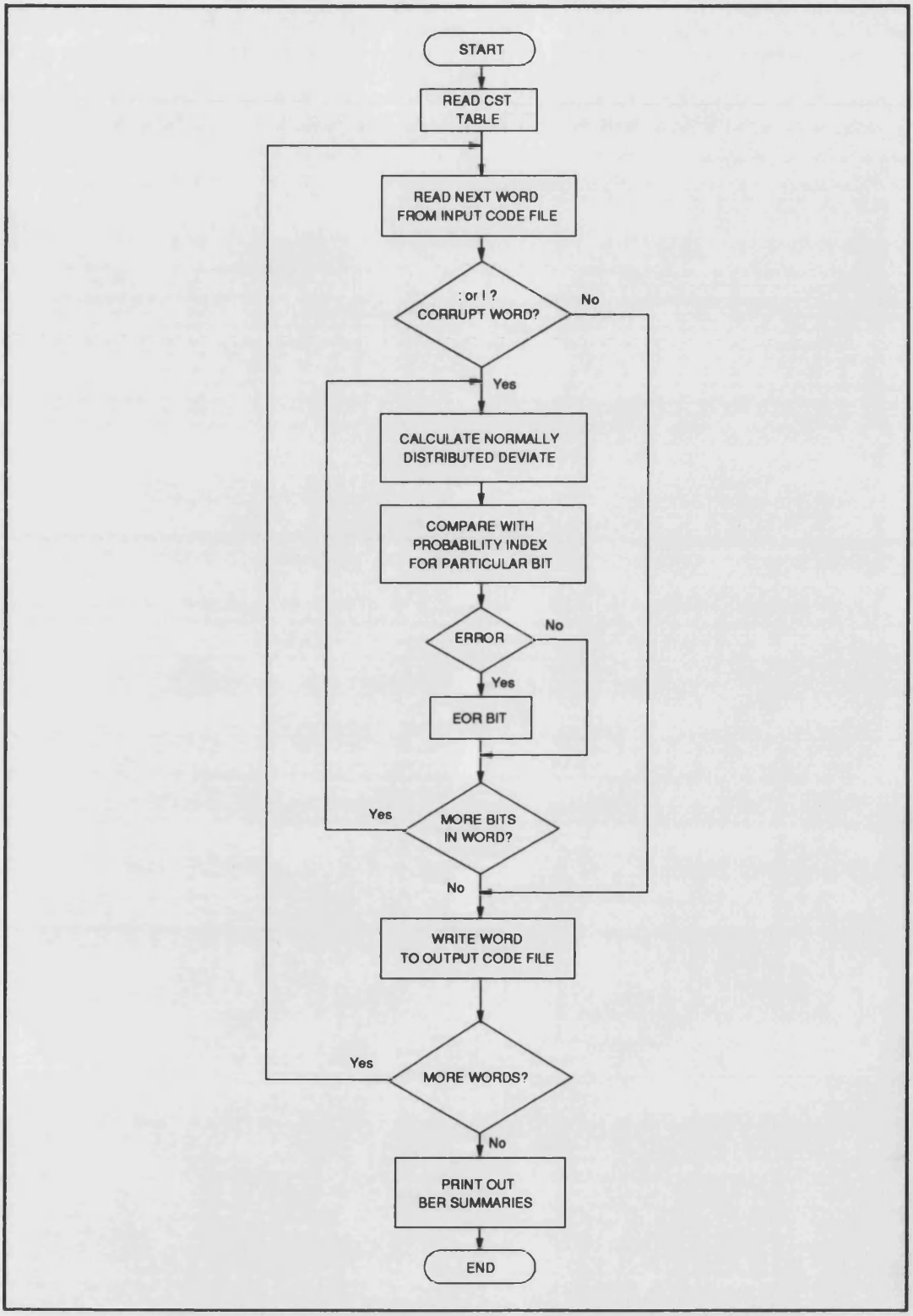

Figure A1.11 *Flowchart of the Mobile Radio Channel Simulator.*

the probability index of the particular bit in question. If this requires an error, the bit is exclusively ORed. Note to ensure a gaussian distributed error distribution, this decision process must consider addition of gaussian noise to the original bit and determine whether it remains correct [A1.10]. When all the bits of a parameter have been tested, the parameter is written to the output code file. The process is repeated for all the parameters of the current frame and then the whole process is repeated for the remainder of the code file.

A1.5 References

A1.1. Loughborough Sound Images Ltd. "TMS 320C25 PC Board User Manual."

A1.2. Ariel Corporation "Operating Manual for the DSP-32C Floating point DSP Coprocessor Board for PC Compatibles."

A1.3. Hyperception "Hypersignal Users Manual."

A1.4. "IEEE Recommended Practice for Speech Quality Measurements." *IEEE Trans. Audio and Electroacoustics,* Vol. AU-17, No.3 (Sept 1969).

A1.5. GSM Recommendation: 06.10 "GSM Full Rate Speech Trancoding."

A1.6. Viswanathan, R. and Makhoul, J. "Quantisation Properties of Transmission Parameters in Linear Predictive Systems." *IEEE Trans. on Acous., Speech, and Signal Proc.* Vol. ASSP-23, No.3, (June 1975): 309-321.

A1.7. Crochiere, R.E. and Rabiner, L.R. *Multirate Digital Signal Processing*. Prentice Hall, Englewood Cliffs, NJ, 1983.

A1.8. Jayant, N.S. and Knoll, P. *Digital Coding of Waveforms: Application to Speech and Video,* Englewood Cliffs, NJ: Prentice-Hall, 1984.

A1.9. Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press.

A1.10. Stremler, F.G. *Introduction to Communication Systems*. Addison-Wesley.

Appendix 2

The Speech Database

In one recording session, 32 speakers were recorded, each saying the 20 sentences listed below. They are taken from the Harvard list of phonetically balanced sentences.

1. Oak is strong and also gives shade.
2. Cats and dogs each hate the other.
3. The pipe began to rust while new.
4. Open the crate but don't break the glass.
5. Add the sum to the product of these three.
6. Thieves who rob friends deserve jail.
7. The ripe taste of cheese improves with age.
8. Act on these orders with great speed.
9. The hog crawled under the high fence.
10. Move the vat over the hot fire.

11. A pencil with black lead writes best.
12. Coax a young calf to drink from a bucket.
13. Schools for ladies teach charm and grace.
14. The lamp shone with a steady green flame.
15. They took the axe and the saw to the forest.
16. The ancient coin was quite dull and worn.
17. The shaky barn fell with a loud crash.
18. Jazz and swing fans like fast music.
19. Rake the rubbish up and then burn it.
20. Slash the gold cloth into fine ribbons.

Each recording was digitised using the available speech recording facilities. 5 test records were electronically edited, consisting of all 20 sentences each spoken by a different person and each test record having 10 males and 10 females. Speakers 13 and

19 were not used since 13 made far too many mistakes and 19 spoke very poor english. From all these recordings, the following table shows how the 5 test records were constructed from all the possible sentences.

		1	2	3	4	5	Record
sentence	1	1m	3f	5m	14f	27m	
	2	2f	15m	22f	27m	10f	
	3	7f	20m	25f	23f	17m	
	4	5m	21m	2f	28m	22f	
	5	12m	7f	15m	10f	26f	
	6	8f	25f	24f	18m	14f	
	7	11m	30m	10f	3f	31m	
	8	9f	12m	20m	5m	29m	
	9	4f	8f	17m	30m	18m	
	10	6m	10f	11m	16m	23f	
	11	14f	16m	12m	24f	7f	
	12	21m	27m	30m	8f	32m	
	13	10f	23f	31m	29m	15m	
	14	22f	11m	26f	20m	1m	
	15	18m	6m	21m	12m	16m	
	16	24f	1m	4f	11m	5m	
	17	32m	4f	3f	22f	9f	
	18	25f	9f	8f	17m	24f	
	19	28m	26f	18m	4f	2f	
	20	31m	2f	14f	9f	25f	

The table elements give the speaker number, and the speaker sex (m/f). Each test record was approximately 53 seconds long.

Appendix 3: Adaptive Codebook Delays

The adaptive codebook delays given in the table below correspond to the delay in sample intervals (with 8kHz sampled speech, each interval is equal to 125μs).

Delay	Code	Delay	Code	Delay	Code	Delay	Code	Delay	Code
20.00	0	34.67	52	51.67	103	68.67	154	97.00	205
20.33	1	35.00	53	52.00	104	69.00	155	98.00	206
20.67	2	35.33	54	52.33	105	69.33	156	99.00	207
21.00	3	35.67	55	52.67	106	69.67	157	100.0	208
21.33	4	36.00	56	53.00	107	70.00	158	101.0	209
21.67	5	36.33	57	53.33	108	70.33	159	102.0	210
22.00	6	36.67	58	53.67	109	70.67	160	103.0	211
22.33	7	37.00	59	54.00	110	71.00	161	104.0	212
22.67	8	37.33	60	54.33	111	71.33	162	105.0	213
23.00	9	37.67	61	54.67	112	71.67	163	106.0	214
23.33	10	38.00	62	55.00	113	72.00	164	107.0	215
23.67	11	38.33	63	55.33	114	72.33	165	108.0	216
24.00	12	38.67	64	55.67	115	72.67	166	109.0	217
24.33	13	39.00	65	56.00	116	73.00	167	110.0	218
24.67	14	39.33	66	56.33	117	73.33	168	111.0	219
25.00	15	39.67	67	56.67	118	73.67	169	112.0	220
25.33	16	40.00	68	57.00	119	74.00	170	113.0	221
25.67	17	40.33	69	57.33	120	74.33	171	114.0	222
26.00	18	40.67	70	57.67	121	74.67	172	115.0	223
26.25	19	41.00	71	58.00	122	75.00	173	116.0	224
26.50	20	41.33	72	58.33	123	75.33	174	117.0	225
26.75	21	41.67	73	58.67	124	75.67	175	118.0	226
27.00	22	42.00	74	59.00	125	76.00	176	119.0	227
27.25	23	42.33	75	59.33	126	76.33	177	120.0	228
27.50	24	42.67	76	59.67	127	76.67	178	121.0	229
27.75	25	43.00	77	60.00	128	77.00	179	122.0	230
28.00	26	43.33	78	60.33	129	77.33	180	123.0	231
28.25	27	43.67	79	60.67	130	77.67	181	124.0	232
28.50	28	44.00	80	61.00	131	78.00	182	125.0	233
28.75	29	44.33	81	61.33	132	78.33	183	126.0	234
29.00	30	44.67	82	61.67	133	78.67	184	127.0	235
29.25	31	45.00	83	62.00	134	79.00	185	128.0	236
29.50	32	45.33	84	62.33	135	79.33	186	129.0	237
29.75	33	45.67	85	62.67	136	79.67	187	130.0	238
30.00	34	46.00	86	63.00	137	80.00	188	131.0	239
30.25	35	46.33	87	63.33	138	81.00	189	132.0	240
30.50	36	46.67	88	63.67	139	82.00	190	133.0	241
30.75	37	47.00	89	64.00	140	83.00	191	134.0	242
31.00	38	47.33	90	64.33	141	84.00	192	135.0	243
31.25	39	47.67	91	64.67	142	85.00	193	136.0	244
31.50	40	48.00	92	65.00	143	86.00	194	137.0	245
31.75	41	48.33	93	65.33	144	87.00	195	138.0	246
32.00	42	48.67	94	65.67	145	88.00	196	139.0	247
32.25	43	49.00	95	66.00	146	89.00	197	140.0	248
32.50	44	49.33	96	66.33	147	90.00	198	141.0	249
32.75	45	49.67	97	66.67	148	91.00	199	142.0	250
33.00	46	50.00	98	67.00	149	92.00	200	143.0	251
33.25	47	50.33	99	67.33	150	93.00	201	144.0	252
33.50	48	50.67	100	67.67	151	94.00	202	145.0	253
33.75	49	51.00	101	68.00	152	95.00	203	146.0	254
34.00	50	51.33	102	68.33	153	96.00	204	147.0	255
34.33	51								

Appendix 4

Paired Comparison Subjective Quality Testing

At various stages during this research, it was necessary to subjectively compare outputs from speech coders. Two sets of listening tests were conducted, the first corresponding to work from chapters 4,5 and 6 and the second corresponding to work from chapter 7.

Each paired comparison required a listener to compare a sentence coded and decoded by two different speech coders. The two output sentences were presented side by side and test 1 contained 36 comparisons and test 2 contained 30 comparisons. Unknown to the listener, these were arranged in male/female speaker pairs, so test 1 compared outputs from 18 pairs of speech coders, and test 2 compared 15. Table A4.1 shows the multiple choice answer sheet for listening test 1. (Listening test 2 had a similar answer sheet with only 30 answers.) For each pair of sentences, the listener was asked to decide which output sounded better and to indicate on the marking sheet by striking either *first, second* or *no preference*. If the listener could not distinguish between sentences, then he was instructed to strike *no preference*.

Each paired comparison consisted of announcement of the sentence number, followed by the pair of sentences. This was repeated for all the paired comparisons in the test. The announcements were replayed using full bandwidth, whilst the decoded sentences had telecommunications bandwidth. This was found to improve the tests impact. Four speakers, two male and two female were chosen for their clarity.

Paired Comparison Subjective Quality Test

Listen carefully to the 36 pairs of sentences, and indicate which sentence from each pair sounds better, using the multiple choice answer sheet below. Strike through either *first* for the first sentence, *second* for the second, or *no preference* if they both sound the same. Each sentence pair will be announced by number and this will be closely followed by the two sentences. The test examines the quality of different speech coding techniques, and you will hear speech from two male and two female speakers.

1.	first	second	no preference	19.	first	second	no preference
2.	first	second	no preference	20.	first	second	no preference
3.	first	second	no preference	21.	first	second	no preference
4.	first	second	no preference	22.	first	second	no preference
5.	first	second	no preference	23.	first	second	no preference
6.	first	second	no preference	24.	first	second	no preference
7.	first	second	no preference	25.	first	second	no preference
8.	first	second	no preference	26.	first	second	no preference
9.	first	second	no preference	27.	first	second	no preference
10.	first	second	no preference	28.	first	second	no preference
11.	first	second	no preference	29.	first	second	no preference
12.	first	second	no preference	30.	first	second	no preference
13.	first	second	no preference	31.	first	second	no preference
14.	first	second	no preference	32.	first	second	no preference
15.	first	second	no preference	33.	first	second	no preference
16.	first	second	no preference	34.	first	second	no preference
17.	first	second	no preference	35.	first	second	no preference
18.	first	second	no preference	36.	first	second	no preference

Figure A4.1 *Multiple Choice Answer Sheet for Listening Test 1.*

1.	GSM	GSM-LTP	NP	19.	SFC-SEV	CELP1G	NP
2.	69	9	22	20.	82	9	9
3.	1%	1%		21.	PFC-SEV+LTP	CELP1G+LTP	
4.	63	14	23	22.	25	48	27
5.	2%	2%		23.	PFC-SEV	CELP1G	
6.	35	32	33	24.	79	11	10
7.	SEV4	CELP4		25.	SFC-N	SFC-A	
8.	49	32	19	26.	48	25	27
9.	11LTP-SEV	GSM		27.	SFC-N	SFC-B	
10.	44	26	30	28.	40	19	41
11.	CELP3G+LTP	CELP3T+LTP		29.	SFC-A	SFC-B	
12.	20	45	35	30.	21	41	38
13.	CELP3G	CELP3T		31.	SFC-N 0.5%	SFC-A 0.5%	
14.	20	36	44	32.	0	100	0
15.	Pure-SEV+LTP	CELP1G+LTP		33.	SFC-N 0.5%	SFC-B 0.5%	
16.	16	73	11	34.	0	99	1
17.	SFC-SEV+LTP	CELP1G+LTP		35.	SFC-A 0.5%	SFC-B 0.5%	
18.	33	33	34	36.	27	24	49

Table A4.1 *Results of Listening Test 1.*

Tables A4.1 and A4.2 lists the overall results of both paired comparison subjective quality tests 1 and 2 respectively, the nomenclature used to describe the speech coders tested is that used within this thesis. Percentages quoted correspond to the nominal BER.

Two problems were identified within the subjective testing, firstly the testing could be improved by having a number of training sentences included in the test. This would enable the listener to accustom himself/herself to the sentence comparisons. In these tests, this was accomplished by playing the first few sentences as training sentences

1.	GSM	GSM-PE	NP	17.	Par 0.5%	Ser/Par 0.5%	NP
2.	23	63	14	18.	32	49	19
3.	CELP3G-EW	CELP3G		19.	CELP1G	Ser/Par-N	
4.	22	59	19	20.	21	48	31
5.	SEV-G	SEV-T		21.	CELP1G	Ser/Par-B	
6.	46	20	34	22.	39	27	34
7.	Ser	Par		23.	Ser/Par-N	Ser/Par-B	
8.	73	16	11	24.	61	11	28
9.	Ser	Ser/Par		25.	CELP1G 0.5%	Ser/Par-N 0.5%	
10.	41	32	27	26.	34	32	34
11.	Par	Ser/Par		27.	CELP1G 0.5%	Ser/Par-B 0.5%	
12.	19	42	39	28.	27	43	30
13.	Ser 0.5%	Par 0.5%		29.	Ser/Par-N 0.5%	Ser/Par-B 0.5%	
14.	27	43	30	30.	24	47	29
15.	Ser 0.5%	Ser/Par 0.5%					
16.	27	46	27				

Table A4.2 *Results of Listening Test 2.*

then rewinding the tape and then starting the test. The second notable problem was the inability of some listeners to acceptably perform the test. Future testing could be improved if potential listeners were selected by testing of their listening ability.